



北京市“211工程”重点建设学科出版专项



首都师范大学音乐学院译著系列

音乐的数字信号处理

一种计算机听觉的视角

[韩] 朴泰弘 著
邓志勇 译

新加坡世界科学出版公司提供版权

 **SMPH**
上海音乐出版社
WWW.SMPH.CN

北京市“211工程”重点建设学科出版专项



首都师范大学音乐学院译著系列

音乐的数字信号处理

一种计算机听觉的视角

[韩]朴泰弘 著
邓志勇 译

上海音乐出版社

新加坡世界科学出版公司提供版权

图书在版编目 (CIP) 数据

音乐的数字信号处理——一种计算机听觉的视角 / [韩]朴泰弘著;

邓志勇译 - 上海: 上海音乐出版社, 2014.9

ISBN 978-7-80751-808-2

I. 音… II. ①朴… ②邓… III. 音频信号处理 IV. TN912.3

中国版本图书馆 CIP 数据核字 (2014) 第 015623 号

Introduction to Digital Signal Processing - Computer Musically Speaking

Copyright © 2009 by World Scientific Publishing Co. Pte. Ltd. All rights reserved. This book, or parts thereof, may not be reproduced in any form or by any means, electronic or mechanical, including photocopying, recording or any information storage and retrieval system now known or to be invented, without written permission from the Publisher.

Simplified Chinese translation arranged with World Scientific Publishing Co. Pte Ltd., Singapore.

书 名: 音乐的数字信号处理——一种计算机听觉的视角

著 者: [韩]朴泰弘

译 者: 邓志勇

出 品 人: 费维耀

责任编辑: 段劲楠

封面设计: 何 辰

印务总监: 李霄云

出版: 上海世纪出版集团 上海市福建中路 193 号 200001

上海音乐出版社 上海市绍兴路 7 号 200020

网址: www.ewen.cc

www.smph.cn

发行: 上海音乐出版社

印订: 上海市印刷十厂有限公司

开本: 787×1092 1/16 印张: 19.5 图、文: 312 面

2014 年 9 月第 1 版 2014 年 9 月第 1 次印刷

印数: 1 - 2,000 册

ISBN 978-7-80751-808-2/J · 743

定价: 48.00 元

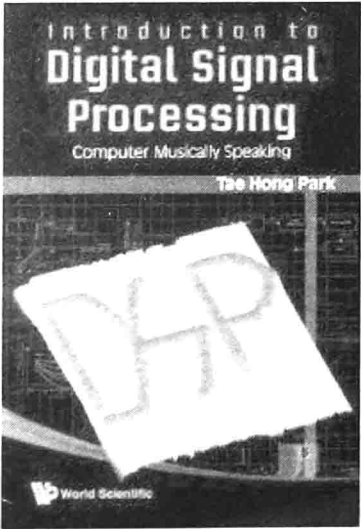
读者服务热线: (021) 64375066 印装质量热线: (021) 64310542

反盗版热线: (021) 64734302 (021) 64375066-241

郑重声明: 版权所有 翻印必究

著者简介:朴泰弘(Tae Hong Park,[韩]),出生于奥地利维也纳。2000 年在美国达特茅斯大学获电子音乐硕士学位,随后在美国普林斯顿大学获音乐学博士学位。时任纽约大学音乐系助理教授,ICMC/ICMA(国际计算机音乐学会)主席和 AES(国际音频工程学会)分会秘书长,曾任美国杜兰大学音乐系助理教授及音乐科技与研究中心主任,纽约大学音乐系副教授,主持过音乐特征调制系统 FMS、新形态音乐研究等多项科研项目,多次在 ICMC、《电脑音乐杂志》(Computer Music Journal)等国际重要会议和期刊上发表学术论文,出版过多部专著,在 Santa Fe 国际电子音乐节、我国中央音乐学院主办的国际电子音乐节等知名音乐会上展演过多首电子音乐作品,是电子音乐、计算机音乐学及音乐声学领域的国际知名青年学者之一。

译者简介:邓志勇,广西柳州人,通信与信息系统专业工学博士,建筑学硕士及舞台科技艺术专业硕士,首都师范大学音乐学院讲师,硕士生导师,美国杜兰大学人文学院音乐系访问学者,英国谢菲尔德大学访问学者,I-INCE 美国国际噪声控制工程学会副会员,在国际学术会议上发表论文 10 余篇,主持、参与多项国家级和省部级科研项目,曾主持和参加北京五棵松奥运篮球馆、上海世博会城市之窗主题馆和深圳交响团音乐厅等重要观演建筑的扩声系统及建筑声学设计。研究领域包括计算机音乐、音乐声学 and 声景学,对 Sound Design 及舞台剧也颇感兴趣。



原著封面简介:原书封面为 D、S 和 P 三个大写字母的动态光谱图。对应的声音由 MATLAB® 代码(bookCover.m)采用滤波器生成,并绘制出了三维光谱图,该源程序可在以下网站下载,<http://music.princeton.edu/~park/dspBook>。

原著评论

在这本书中,朴泰弘建立了一座连接数学声学工程世界与计算机音乐世界的坚固桥梁。无论这座桥上的旅人将会驶向何方,他们都会得到清晰而富于启发性的经验。

保罗·兰斯基
普林斯顿大学“威廉·舒贝尔·科南特”音乐教授
新泽西州

朴泰弘博士的书以独特的跨学科视角和严谨性,介绍了音频信号处理的基本原理,带来了令人兴奋而富有深度的观点和具有创造性的应用。

艾罗罗·奥瑞芬
加利福尼亚州旧金山市音频工程师,作曲家

朴泰弘关于音乐数字信号处理的新书犹如一颗真正的宝石,每一个概念的论述都提供了许多生动的图片和容易理解的源代码。我期待着把这本书用做教材,相信学生一定会喜欢它。

格奥尔格·埃斯尔
柏林工业技术大学,德国电信实验室高级研究员

序

我与译者是在 2003 年春天认识的。在华南理工大学的红楼里,作为建筑声学研究生的他一见我,便聊起他的戏剧经验和对舞台音响的兴趣。同年,他来到香港演艺学院成为了我的学生,主修的是音响设计及音乐录音专业。当时,香港演艺学院的专业学习没有预设的课程,所有课程都是我与他共同度身打造的。他的毕业作品是大型音乐剧《罗密欧与朱丽叶》的音响系统和音效配乐设计,这部作品的成功把他带进了音响设计和音乐信号处理领域。之后,他在北京从事了一段时期的音响设计工作,继而重返校园,成为了中国传媒大学研究音乐声学的博士生。后来,他在首都师范大学音乐学院做了老师,还曾前往美国爵士乐的发源地新奥尔良进行学习。

如今,令人欣慰的是,他在自己坚实的声学研究基础上,探讨音响艺术与工程的奥妙,将声学与音响相结合,走进了声音与音乐的深层研究中,并在著者——青年电子音乐家朴泰弘先生的帮助下,为我们翻译了这本有关音乐数字信号处理的著作,这是音响人和音乐工程的专业人士以及众多音乐家梦寐以求的好事。这一名为《音乐的数字信号处理——一种计算机听觉的视角》的中译本是有关数字信号处理在音乐音频信号及计算机音乐应用方面的导论,内容包括人耳听觉及其局限性、时域信号处理、正弦波、线性时不变系统、频率响应、滤波器、频域与傅立叶变换、谱分析及语音合成等八个部分,每一部分都有详细的图表、MATLAB[®] 源代码和饶有趣味的乐例分析,让读者从数学、物理和音乐等多个层面感受音乐数字信号处理的魅力。

特别向音频及音乐工程的从业人士、研究者和学生推荐这一译著。

李永荣

香港演艺学院教授

上海音乐学院客座教授

2013 年 3 月 9 日于香港

译 序

初次见到该书的著者朴泰弘(Tae Hong Park)先生,是在2008年中央音乐学院主办的国际电子音乐节上,他名为“音乐特征调制系统”(Music Features Modulation System)的主题报告和那首满怀思乡情结和怀旧情绪的声景音乐作品(Soundscape composition)《48-13-N-16-20-O》让我印象深刻。后来在吴雯学友的帮助和联系下,我们得以相识,聊得甚为投机。2010年,在首都师范大学青年教师境外研修基金的支持下,我来到美国新奥尔良杜兰大学(Tulane University)音乐系音乐科学与技术研究中心跟随朴先生进行访学研修,关注于声景音乐创作和音乐的数理与心理声学分析,并有幸拜读了他这本关于音乐数字信号处理的著作。该书将科学与艺术相融合视野,理性、幽默和激情并重的行文风格,以及书中那些经典乐例所构成的一部微缩的电子与流行音乐史,激发了我将这本对音乐爱好者来说不可多得的读本译成中文的想法,这也得到了朴先生的欣然应许。该书的翻译在朴先生的指导下从2010年12月开始至2011年6月完成初稿,后又经2012年和2013年的两次修订才交付出版。此次中译本的内容包括人耳听觉及其局限性、时域信号处理、正弦波、线性时不变系统、频率响应、滤波器、频域与傅立叶变换、谱分析及语音合成等八个部分,共九个章节,保留了原书的全部章节内容、MATLAB[®]源代码和所有的音乐实例。本书既可以作为音频音响工程、音乐声学、计算机音乐学、声音信号处理等领域的研究者和从业人员的参考书,也可以作为电子音乐家和众多计算机音乐爱好者及音乐制作者的专业技术手册,同时还可以用作相关专业本科生和研究生的课程教材。

追忆这本译著的创作足迹,不仅留在具有百余年历史的杜兰校园,还留在承载着爵士乐荣耀的法国老街,马克·吐温笔下的海港,铭刻着《飘》的记忆的橡树庄园,庄严的洛约拉大天主堂,以及静谧而不再熙攘的密西西比河畔。抚稿而思,感慨良多。音乐学科本身就犹如新奥尔良这座历经岁月和灾难洗礼的城市一样,既古老沧桑而又富有重生后的勃勃生机。

我衷心感谢首都师范大学音乐学院和上海音乐出版社提供出版基金,使该书得以面世,感谢香港演艺学院的李永荣先生为本书作序,感谢

最初引领我进入声学领域的华南理工大学的吴硕贤院士和赵越喆博士,感谢为我的信号处理专业打下坚实基础的、在诸多领域为我打开广泛视野的中国传媒大学的孟子厚教授,感谢在翻译和文学创作上给我诸多鼓励和指导的黄鹤教授,感谢张健博士在信号处理研究方面对我的鼓励和多年来对我的关心与照顾,感谢刘爱利博士给我留美访学的信心,感谢吴雯和李智也学友对我在美访学期间的帮助和照顾,感谢马克西姆·萨马洛夫(Maxim Samarov)先生在作曲、指挥、音乐创作与分析方面的指导,感谢王岱威和李昕欣同学为本书校对工作做出的贡献,特别再次感谢朴泰弘先生对我在音乐信号处理及电子音乐创作、演出专业方面的指导,感谢所有支持本书翻译及出版的朋友们。

末了,我不会忘记感谢我的父亲邓小华,母亲刘玉英,妹妹邓艳娜和弟弟邓志军以及他那个可爱的女儿——我的侄女。他们多年来对我从事科学研究给予了极大的理解和无条件的支持,尽管至今他们仍然不清楚我为何在十年前放弃建筑师职业来做这些研究。

成书仓促,才学尚浅,不免有错漏疏忽之处,还望各位同行不吝赐教,批评指正。

邓志勇

2013年5月20日于北京杏园

原 序

这是一本旨在为有兴趣以音频信号为侧重点学习数字信号处理(DSP)的读者提供参考的论著。这本导论性很强的著作涵盖了数字信号处理的基本领域,包括与采样、滤波器、声音合成算法、声音效果器和时域及频域分析方面的重要理论与应用,以及以信号处理为中心的诸多专题。本书以 DSP 理论为出发点,通过与概念及数学原理相关的音乐和声音实例,来阐释音频和音乐信号,从而提供一个有吸引力且直观的方式,让所提供的材料变得熟悉而易于接受。对于本书的所有章节,我尽可能按照 L^3 (看、听、学)教学模型进行组织——多年来,我的哥哥早已使这种教学模式深深植根于我脑海之中。书中所提及的大量概念和图表及其相应的 MATLAB[®] 代码,可以从网址 <http://music.princeton.edu/~park/dspBook> 中下载(详情参见附录)。本书在论述过程中,对于重要的理论和观点的成因(即“为什么”的问题)给予了特别关注,尤其是当新的材料被第一次提到的时候。这种方法通常很少运用在标准的工程书籍中,有时在很大程度上只是罗列非常抽象的概念,很少以实际情况和应用为参照。同时,这本书也尝试填补谱分析的另一空白——即被一些与信号处理有关的非工程类书籍所忽略的、无法深入的数学层面的阐述。本书的目标关注于“为什么”和“怎么做”,其最终目的是帮助读者学习和理解数字信号处理的艺术。至于读者的数学背景要求并不算高,书中大多数概念属于有限的代数知识范围内。因此,本书适用于广泛的读者群,包括音乐家、作曲家、工程师、计算机科学家、程序员和相关学科的本科生和研究生。

本书每章的结构都包括概念的介绍,数学基本原理,可操作的实例,大量的图表和结束部分与所涉及内容相对应的乐例分析。全书以大家熟悉的正弦波引入,以时域信号处理开篇,同时还介绍了声学的基本知识和人耳听觉的局限性。这一设计能让读者理解如何使计算机在不牺牲音质的前提下,以容量更小的数字化信号来表示和重现模拟的声音。用量化与比特率进行人工采样,且称之为 CD 音质的技术标准作为数字音频领域中重要概念的出现铺平了道路。这些概念包括:声音的 ADSR 瞬态特性,窗口化,RMS 包络检测,波表合成和采样率变换。相关的乐例有皇后乐队的

《给尘土的另一个吻》(*Another One Bites the Dust*)和威廉·斯克托尔德(William Schottstaedt)的《巨物》(*Leviathan*)中所运用的时域时间延展和压缩效果。

在这本书中,尤其是在开始阶段,DSP 知识以一种潜意识的方式介绍给读者,即在 DSP 和数学的语境中,关注并呈现一些有趣而且令人感兴趣的声​​音合成与分析实例。例如,一开始我们引入正弦波、调幅、调频等概念,并用拍声技术(clapping technique)来解释脉冲响应,从而为接下来介绍音频卷积的理论和应用做准备。到了书的中部,读者们能够以足够的信心进入信号处理的核心部分进行学习——相关各专题如差分方程、频率响应、 z 变换和滤波器等。通常,DSP 的相关实例将附在每个章节的末尾介绍,同时各专题的 MATLAB[®] 源程序,如梳状滤波器及其弹拨乐器的物理建模等也有所涉及。全书的最后一章,我们将全面进入对信号频域分析的探索中,关注于傅立叶变换(重点放在离散傅立叶变换 DFT 上)及其在计算机音乐各个领域的应用。此时,我们还会回溯本书开始处介绍的概念,如过采样、减采样、抽取、谐波失真和抖动以及奈奎斯特定理,进而从不同的角度巩固我们对这些非常基本和重要的信号处理知识和概念的掌握。事实上,这一章是在重新审视以前所学的知识 and 那些在全书中反复出现的论题。希望这能进一步帮助我们加强对本书所提到的原理的理解。最后一章还讨论了经典声码器算法,并对数字音频与计算机及电子音乐领域的研究专题进行了概述。

朴泰弘

2009 年 5 月于美国路易斯安那州新奥尔良市

著者致谢

在卡特里娜飓风期间,我开始了这本书的写作。那个学期我们不得不逃离新奥尔良——在 2005 年的秋天,那座城市被洪水淹没。我带着我的妻子和我们的猫逃到了印第安纳州西拉斐特市(West Lafayette),因为我们知道在很长一段时间里不能回家了。在那个动荡的时期,我开始与世界科学出版公司(World Scientific Publishing Co.)接洽,正如他们所说,这是一段历史。

对我来说,写这本书是一个有趣的、快乐的、要求很高的、非常有价值的事情。不用说,我要对那些直接或间接帮助我完成这本书的人给予极大的感谢。

首先,我要感谢我的父亲朴董圭(Dong Kyu Park),母亲杨钟淑(Chung Suk Yang)和两个兄弟朴奇弘(Kihong Park)和朴宰弘(Jae Hong Park)。他们一直在那里无条件地支持我。

我也要感谢李智也(音,Zhiye Li)和特拉维斯·沙尔(Travis Schar),他们对修改本书许多错误做出了巨大贡献。特别感谢艾罗罗·奥瑞芬(Iroro Orife)、格奥尔格·埃斯尔(Georg Essl)和泰德·科菲(Ted Coffey)提出了许多有见地的意见和批评,感谢你们对书中材料的校对(当你们写书需我帮助时尽请吩咐)。我相信书中还有许多或大或小的错误。不过,我认为在书稿交付出版之前,如果我的朋友已经更正了其中许多错误,定使我更加放心。话虽如此,读者对本书的任何意见、建议和勘误报告,均可在网址<http://music.princeton.edu/~park/dspBook>上留言。

我还想感谢我的老师,约翰·阿普尔顿(Jon Appleton)、佩里·库克(Perry Cook),查尔斯·道奇(Charles Dodge)、保罗·昆斯(Paul Koonce)、

保罗·兰斯基(Paul Lansky)、拉里·波兰斯基(Larry Polansky)和丹·楚门(Dan Trueman)对我在研究生期间的影响、教导和训诫。

还要感谢易查努(音, Chanu Yi)和世界科学出版公司的同胞——孙春光(音, Chunguang Sun)和切尔西·秦(音, Chelsea Chin)。

最后,最深的感谢献给我的妻子安贞焕(Kyoung Hyun Ahn),献给她无尽的支持、鼓励和爱。好了,要回到工作中去了。

朴泰弘

2009年5月于美国路易斯安那州新奥尔良市

目 录

著者简介、译者简介、原著封面简介 i

原著评论 ii

序 iii

译序 iv

原序 vi

著者致谢 viii

第 1 章 声学,人耳听觉的局限性及采样 1

1. 引言 1

2. 正弦波 1

3. 人耳听觉及其局限性 3

3.1 时长 3

3.2 音高 3

3.3 振幅与声级 4

3.4 听觉掩蔽 7

4. 采样:离散化的艺术 9

4.1 采样定理 12

4.2 频率混叠 13

5. 量化与脉冲编码调制(PCM) 17

信噪比(SNR)和量化信噪比(QSNR) 19

6. DC 直流分量 20

7. 失真与方波 21

抖动 22

8. 乐例 25

参考文献与扩展阅读 26

第 2 章 时域信号处理(一) 27

1. 引言 27

2. 振幅包络与 ADSR 结构 27

3. 波表合成 29

4. 时窗分析,RMS 与振幅包络 31

4.1 时窗分析的进一步讨论 31

4.2 RMS 与振幅包络 34

5. 时域内的基频提取方法 36

5.1 过零率	36
5.2 自相关函数	40
5.3 互相关函数	44
6 采样率变换	44
6.1 过采样	44
6.2 减采样	45
7. 叠加	47
叠加算法的局限及其解决办法	48
8. 乐例分析	49
参考文献与扩展阅读	50
第3章 时域信号处理(二)	51
1. 引言	51
2. 粒子合成	51
2.1 粒子合成的基本参数	52
2.2 异步粒子合成	53
2.3 移调与时间延展及压缩	54
2.4 用粒子合成进行声变形	55
3. 振幅失真与波形整形	55
3.1 动态压缩	56
3.2 失真	57
3.3 动态扩展	58
3.4 磁带饱和度失真	58
3.5 波形整形合成	59
4. 几个相似的时域 DSP 效果	61
4.1 等功率平移	61
4.2 延迟	62
5. 乐例分析	63
参考文献与扩展阅读	64
第4章 正弦波	65
1. 引言	65
2. 正弦曲线	65
3. 虚数,复数与欧拉公式	66
欧拉公式	68
4. 正弦波调制技术(1):调幅	68
4.1 拍频	68
4.2 振幅调制与环形调制	70
4.3 振幅调制(AM)	70

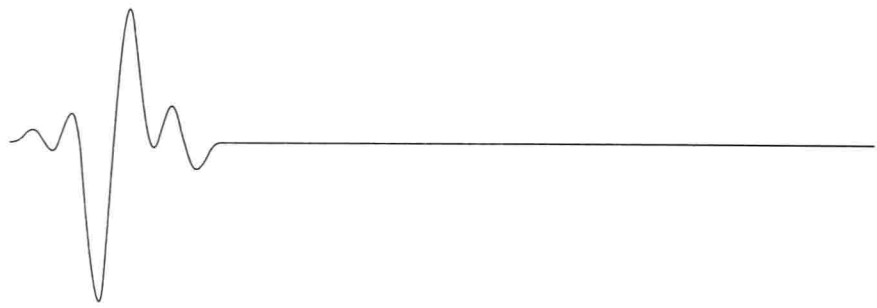
4.4 环形调制	71
5. 正弦波调制技术(2):调频	73
5.1 调频:旁频与贝塞尔函数	74
5.2 调制指数	77
5.3 频率调制的控制参数	78
6. 乐例分析	78
参考文献与扩展阅读	80
 第 5 章 线性时不变系统	81
1. 引言	81
2. 差分方程:移动平均算法	81
2.1 因果性	83
2.2 差分方程的一般形式	84
3. 线性时不变 LTI 系统	84
3.1 线性:伸缩不变性与可加性	85
3.2 时不变性:时移不变	85
3.3 数字信号处理中 LTI 系统的重要性	86
4. 脉冲响应	87
4.1 有限脉冲响应(FIR)与无限脉冲响应(IIR)	89
4.2 稳定性与 IIR 系统	90
5. 卷积	91
必须知道的卷积性质	92
6. 系统流程图与数字系统框图	93
7. 乐例分析	95
 第 6 章 频率响应	96
1. 引言	96
2. 频率响应	97
2.1 $H(e^{j\theta})$ 的特征与性质	98
2.2 频率响应的其他性质	103
3. 相位响应和相位失真	105
3.1 相位延迟	107
3.2 线性与相位	109
3.3 相位响应与连续相位	110
3.4 群延迟	111
4. 神奇的 Z 变换	114
4.1 Z 变换的含义(1)	114
4.2 Z 变换的含义(2):极点和零点	115
4.3 Z 变换的含义(3):单位圆与 Z 平面	116

4.4 更“复杂”的系统	119
5. 收敛域 ROC	120
5.1 因果系统的收敛域	123
5.2 复合因果系统	124
5.3 收敛域 ROC 小结	124
6. 稳定性与单位圆	125
7. 逆 Z 变换	126
7.1 长除法	126
7.2 泰勒级数展开法	128
7.3 曲线积分与留数定理	129
8. 有用的 MATLAB® 工具包	131
9. 乐例分析	131
参考文献与扩展阅读	132
第 7 章 滤波器	133
1. 引言	133
2. 低通、高通、带通和带阻滤波器	133
2.1 滤波器的设计技术参数	135
2.2 通带、阻带与过渡带	136
2.3 截止频率	136
2.4 滤波器的阶数、锐度和纹理	136
2.5 MATLAB® 的滤波器设计工具	144
3. 滤波器实例	147
3.1 减法合成与滤波器	147
3.2 双二次滤波器与颤音滤波器	149
3.3 梳状滤波器	153
3.4 弦振动与驻波	161
3.5 物理建模合成与弹拨乐器模型	163
3.6 相位在滤波器中的应用	169
4. 乐例分析	181
参考文献与扩展阅读	182
第 8 章 频域与傅立叶变换	184
1. 引言	184
2. 加法合成	184
3. 傅立叶变换	186
4. 离散时域傅立叶变换与离散傅立叶变换	187
4.1 DFT 的幅值、相位与其他基本性质	192
4.2 DFT 的时域分辨率与频域分辨率	193

5. 短时傅立叶变换 STFT	194
6. 零填充	195
7. 频率混叠的再次讨论	197
8. 另一个视角的减采样与过采样	199
8.1 减采样	199
8.2 过采样	201
9. 时窗的频域分析	203
9.1 矩形窗	206
9.2 汉恩窗	207
9.3 汉明窗	207
9.4 布莱克曼窗	208
9.5 切比雪夫窗与凯瑟窗	208
9.6 时窗的其他问题	210
10. 快速傅立叶变换	213
11. 卷积的再讨论	214
循环卷积与时域混叠	215
12. 抖动	217
13. 光谱图	218
14. 傅立叶变换性质小结	219
15. MATLAB [®] 与傅立叶变换	220
16. 乐例分析	221
参考文献与扩展阅读	222
 第 9 章 频谱分析、声码器及其他问题	223
1. 引言	223
音乐信号与重要术语	223
2. 频谱分析	224
2.1 长时平均频谱	224
2.2 对数坐标与线性坐标	225
2.3 频谱的峰、谷和频谱包络	227
2.4 基频及其谐波的提取	230
3. 声码器	235
3.1 通道声码器	236
3.2 线性预测编码	246
3.3 LPC 系数的计算	249
3.4 相位声码器	250
4. 计算机音乐的若干研究专题	265
4.1 显著特征量的提取	265
4.2 音乐信息检索 MIR	269

4.3 特征调制合成系统 FMS	271
5. 乐例分析	273
参考文献与扩展阅读	274
附录	278
1. 伸缩还是不伸缩,是一个问题	278
1.1 等程尺度	278
1.2 调音	278
1.3 Bark 尺度	278
1.4 Mel 尺度	279
2. 本书的 MATLAB® 程序	280
参考文献与扩展阅读	284
索引	285

第 1 章 声学,人耳听觉的局限性及采样



1. 引言

从本章开始,我们将通过大家已经熟知的话题进入到数字信号处理世界的旅程中。我们首先简略介绍正弦波的特性,随之介绍声学中重要的基础性概念,讨论人耳的听觉特性及其各种有趣的局限性,并引出模拟信号数字化的概念。正如我们在接下来的章节中所看到的那样,将模拟信号(如声波)进行数字化的精度,在很大程度上取决于人耳听觉系统的局限性,因此在本书中“局限性”(limitation)一词仅相对于数字系统而言。人类的耳朵、眼睛和其他感觉器官并非具有无限的分辨率。虽然这是人类感知系统的不足,但可以在实践中加以利用,如这种感知的局限性可以大大降低大脑和计算机处理信息时的冗余量。

2. 正弦波

之所以在一开始就介绍正弦波是基于以下原因:一是在计算机上容易实现、重放和重现正弦波的波形;二是正弦波也许是人们最为熟悉的声音;三是正弦波包含了振幅、频率和相位等声音的基本特征。

如图 2.1,我们也许还记得在物理实验室或一些科教电视节目中对振荡模式所作的讲解——用一支笔在上下做垂直往返振荡运动的同时向左做水平匀速运动,这样就能画出下述图形。经过一段时间之后,我们会在这张纸上得到一个振荡轨迹——正弦波形。这种振荡模式也可以用图 2.1 中左图所示的单位圆来表示,笔沿该单位圆做逆时针方向的匀角速度运动,笔在圆上的位置用弧度表示,振荡位于峰值时,其在单位圆上对应的弧度值为 $\pi/2$,振幅为 1.0。如式 2.1 和表 2.1,正弦振荡又可以用三个特征参数来描述,即振幅、频率和初相位。

$$y(t) = A \cdot \sin(2 \cdot \pi \cdot f \cdot t + \phi) \tag{2.1}$$

振幅是一个归一化到 ± 1.0 之间的任意实数。一秒钟内振荡的次数定义为赫兹(Hz),一个完整的往返振荡运动称为一个周期。初相位 ϕ 则是指笔在单位圆的哪一个位置开始做逆时针方向的正弦振荡运动。如当 $\phi = \pi/2$ rad 或 90 度时,正弦波形会变成一个余弦波形。图 2.2 给出了一个完整周期的正弦波。

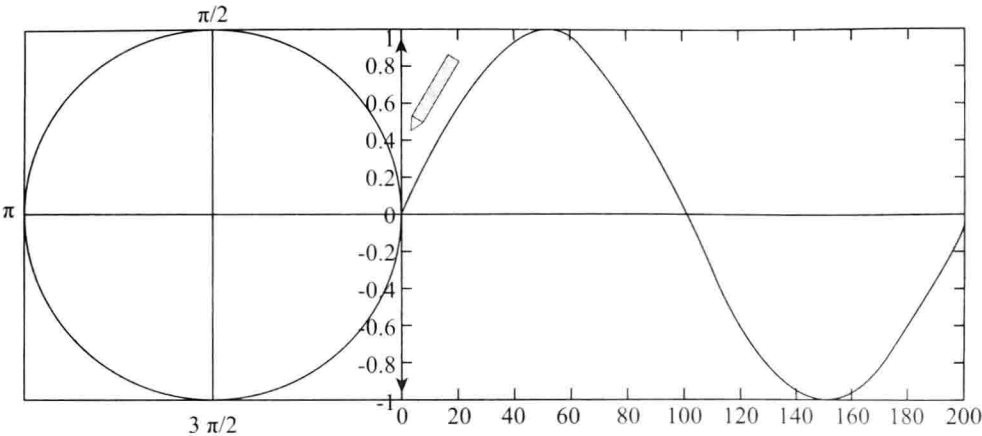


图 2.1 单位圆与正弦波

表 2.1 正弦振荡器的参数

参 数	描 述
A	振幅(在此例中 $A \in [-1, +1]$)
f	频率,单位赫兹(Hz)
t	时间,单位秒(s)
ϕ	初相位,单位弧度(rad)

我希望读者拥有 MATLAB[®] 软件来运行随后的代码,我们将用 MATLAB[®] 源代码来展示 DSP 实例,生成声音和绘制图形。如果没有该软件也不用担心,只要您有编程经验,这些代码中的算法思想将对您的理解有所帮助。如果读者有 MATLAB[®] 软件,则运行如下代码,即可听到一个频率 $f=440\text{Hz}$ (音高等同为音符 A4),振幅 $A=1$,时长为 1 秒,初相位 $\phi=0$ 的正弦波。

```
y = sin(2*pi*440/44100*[0:44099]);
sound(y, 44100)
```

代码 2.1

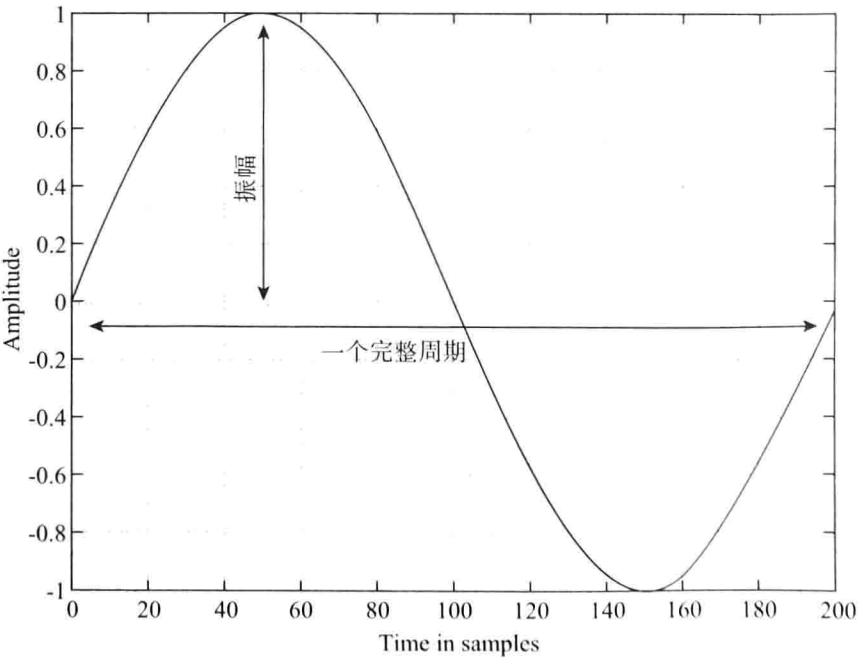


图 2.2 一个周期内的正弦波振幅

3. 人耳听觉及其局限性

声音通常可以用四个基本但很重要的参数来描述,即时长(duration)、音高(pitch)、振幅(amplitude)和音色(timbre)。不论是在自然界中听到的声音,还是在音乐厅中通过扩声系统重放的音乐,又或是在一个宁静的星期一夜晚在家中欣赏你喜爱的轻音乐CD,这四个参数都非常关键。不过,一开始我们或许不能明显感觉到这些参数,正如由于人耳听觉的局限,我们不能听到空气的轻微振动所产生的任何声音一样。这一节我们就将讨论人耳的局限性问题。

3.1 时长

时长,是表征一个纯音或声音事件持续时间的最直接方式,单位为秒、分和时等。乍一看时长这个概念非常简单,但当它与人类感知和心理声学结合起来考虑时,人耳的听觉对时长的感知立刻就变得十分复杂。比如,当你在安静的环境下通过立体声系统聆听一首时长为3分钟的歌曲时,你能感觉到该首歌的长度确实为3分钟。但如果是在背景噪音强度始终与乐曲一致的晚会上聆听同一张CD、同一轨、同一时长的歌曲呢?这首歌的长度听起来也许不会再是3分钟,尽管这两个版本的时长在理论上确实都是3分钟。这一节我们不会深入讨论声音的心理声学效应,只会涉及一些声音领域中可测量的基本概念参数,如音乐或声音事件时长等重要问题。

3.2 音高

音高是对声音周期(periodic)或者说准周期(quasi-periodic)特征的听觉感知。音高越高意味着被感知的频率越高,同理,音高越低意味着被感知的频率越低,频率的单位为赫兹(Hz)。用频率来表示音高时,它与时间 t 的倒数关系如式3.1。

$$f = 1 / t \quad (3.1)$$

可以试着改变频率参数,并跟着MATLAB®代码2.1所生成的纯音哼唱“啊”。当然,并不是所有乐器听起来都有音高。如铃鼓(tambourine)或底鼓本身并没有音高。尽管在理论上底鼓可以调至较高的共振频率,使它听起来好像有了一定的音高。不过在大多数情况下,特别是对流行音乐而言,这种做法并不常见。不一样的是,定音鼓则具有固定的音高。至于底鼓,想象乐队的鼓手水平不高,每次在演唱会中演奏新的歌曲时,都会由于调的变化而重新对整套鼓进行调音,而不只是底鼓。音高通常由物体振动的固有频率,即基频所决定。但有时候并不存在基频,我们也能感知出音高,这种情况称为“基频缺失现象”(missing fundamental phenomenon)。听力正常的人能够听到的声音频率范围大致为20Hz至20,000Hz,但这并不意味着人耳能够感知的音高也对应着同样的频率范围,准确的音高感知频率的上下限因人而异。通常认为200~2,000Hz涵盖了人耳感知频率变化最敏感的区域(Dodge, Jerse 1985),这也是为什么钢琴的音域正好位于27.5Hz~4,186Hz之间的原因之一。当频率升高至超过4,000Hz时,感觉音高会慢慢变化至“高频”,高于实际所对应的音高频率。需要区别的是,当频率升高至接近人耳听觉感知频率上限时,我们会认为存在一个比实际上限音高更高的频率分量,同样的现象在人耳听觉感知频率下限时也存在。当一个脉冲信号的频率从1Hz慢慢提高至30Hz附近时,我们在频域能感知到以下三个现象:

- 1) 在一个较低的频率范围,我们能够数清或基本上数清每秒钟脉冲信号的个数,有时会引发节奏的感觉;
- 2) 当脉冲信号快至已经不能数清时,听觉上就出现了粗糙感;
- 3) 最后随着频率升高至一定的范围,粗糙感会转变为对音高的感知。

此外,两个常见的音高感知理论是位置理论 (place theory) 和听觉神经纤维锁相理论 (phase-locking of nerve fibers)。总而言之,人耳不能感受到所有的音高和频率,人耳能够感受的频率范围有一个模糊的上下限。

3.3 振幅与声级

振幅,即信号的强度,振幅越大声音越大,反之亦然。对声音信号来说,常用分贝 (dB) 作为量度振幅的单位。一组相似而容易混淆的参量也可以用来量度声音和音乐的振幅,它们是声强级 (sound intensity level, 简称 SIL)、声压级 (sound pressure level, 简称 SPL) 和响度 (loudness)。其实,这三个参量在互换使用时往往并不会被混淆。有些混淆的情况也许是因为和时长或时间的概念一样,响度这个心理声学的参数在概念上会被直观地认为是调整我们立体声设备中的音量旋钮 (现在通常是按钮) 来获得更响的声音。实际上可以说“我们所感觉到的声音并非是我们所听到的” (what we hear is not what we hear), 接下来一节就将讨论以上所提及的这些有关振幅的参量。

3.3.1 声强级 SIL

设有一球状声源 (半径为 r_{small}) 的表面被无数个半径较小的扬声器所覆盖,在同一介质中 (如空气) 各向声能辐射特性相同。同时,再设该球状声源外有一个更大的球——半径为 r_{big} 同球心的球,这样我们可看到当球状声源发声时,声音会击打外球的内表面。则将该声源在单位时间内,在与其声能辐射方向垂直的单位面积所通过的平均声能定义为声强,量度单位为 w/m^2 。我们也有这样的经验,距离声源越远,声音将变得越柔和。这就是下式所表达的球状声源声强与距离的平方反比关系:

$$I = \frac{P}{4\pi r^2} \tag{3.2}$$

其中 P 是声源的声功率, r 是接收点与声源的距离。该式的基本意思是,接收点与声源的距离 r 增加,声强随之呈指数衰减,而不是线性的。声强级 (SIL) 的定义如式 3.3。

$$dB_{SIL} = 10 \cdot \log_{10}(I_1/I_0) \tag{3.3}$$

单位“分贝”一词来自于电话的发明人亚历山大·格雷厄姆·贝尔 (Alexander Graham Bell) 的名字,贝尔的定义如下:

$$bel = \log_{10}(I_1/I_0) \tag{3.4}$$

式 3.3 与 3.4 的区别在于乘了一个系数 10,即分贝是贝尔的十分之一。分贝这一单位是后人在贝尔单位前加上一个十分之一发展出来的,目的在于增强贝尔的易用性和可靠性。

在式 3.3 与 3.4 中, I_1 是待测声强, I_0 是参考声强,是一个以功率来量度的声强听觉阈限,即把一个频率为 1 kHz 的纯音的声强由零慢慢提升至该阈限 (I_0) 时,人耳恰能听到。选择 1 kHz

信号的原因是基于弗莱彻-曼森(Fletcher-Munson)的纯音等响曲线——假设人耳对所有频率的声音在强度上的反应是一样的,需要选择一个有代表性的纯音作为参考。显然在不同的场合参考声强 I_0 也可以取其他常数,但对声音而言,声强的人耳听觉阈值定义为:

$$I_0 = 10^{-12} \text{ Watts/m}^2 \quad (3.5)$$

声强级的一个重要特性是,它是一个与参考声强的能量比,而不是一个绝对值。那就是说,如果需要将声强级 SIL 提高一倍,如由 10 dB_{SIL} 提高至 20 dB_{SIL} ,则需要将 I_1 与 I_0 的比值提高至其平方值。为了简单明了,又设 $I_0 = 1$,则 $10 \cdot \log_{10}(2 \cdot 10) = 13.01 \text{ dB}$,声强级只提高了 3.01 dB ,而不是 10 dB_{SIL} 的两倍。在此例中,如果将声强提高 10 倍(即扩大 I_1 与 I_0 的比),声强级 dB_{SIL} 将提高至 20 dB :

$$10 \cdot \log_{10}(10 \cdot 10) = 10 \cdot \log_{10}(10^2) = 20 \cdot \log_{10}(10) = 20$$

3.3.2 声压级(SPL)

另一个常用的量度声音强度的参量是声压级(SPL),其单位是 dB_{SPL} ,定义如式 3.6。

$$\text{dB}_{SPL} = 20 \cdot \log(A_1/A_0) \quad (3.6)$$

由上式可知,声压级与声强级定义的不同之处是,声压级定义中对数前的系数由 10 换成了 20,而参数名称也做了改变——即改成了压强幅值 A_1 和 A_0 之比。 A 单位为帕斯卡(Pa),用来表示为空气中的声压和人耳对 $1,000\text{Hz}$ 纯音的声压听觉阈值 $A_0 = 2 \times 10^{-5} \text{ N/m}^2$ 。上述定义式就来自于振幅的平方与声强成正比,即 $I \propto A^2$,则有式 3.7。

$$20 \log_{10} \frac{A_1}{A_0} = 10 \log_{10} \left(\frac{A_1}{A_0} \right)^2 = 10 \log_{10} \frac{A_1^2}{A_0^2} \quad (3.7)$$

图 3.1 给出了典型的的声音事件声级,该图同时也显示了数字系统和数字信号处理中可以利用的人耳听觉在强度感知方面的局限。

假如我们曾经遇到或听到音频工程师提过一些常见的 dB 声级,如 $0 \text{ dB}_{SPL} (A_1 = A_0)$ 和 $6 \text{ dB}_{SPL} (2 \cdot A_1/A_0)$,现在我们理解它们的含义了。

3.3.3 最小可觉差别阈限(JND)

最小可觉差别阈限(Just noticeable difference,简称 JND),对于声音强度来说,定义为能够被听者所察觉的声音振幅的最小差别(而对于声音频率同样是指最小可觉差别)。最小可觉差别阈限的存在意味着人耳听觉有更多的局限性。通常用纯音来测量最小可觉差别阈限,实际阈限值取决于使用哪个频率的纯音以及该音的分贝值。一般认为最小可觉差别阈限值为 1dB 。图 3.2 为 1 kHz 纯音和它的最小可觉差别阈限特征曲线,这种特征也许就是现在的立体声系统中音量控制钮设计为离散按钮而不是连续旋钮的原因。

3.3.4 等响曲线

人耳不是对所有频率都有良好的听觉感知的。例如,即使我们不改变功率放大器的振幅或者输出音量,当听音量一样的 100Hz 正弦波和 1kHz 正弦波时,我们也不会觉得它们有同样的响

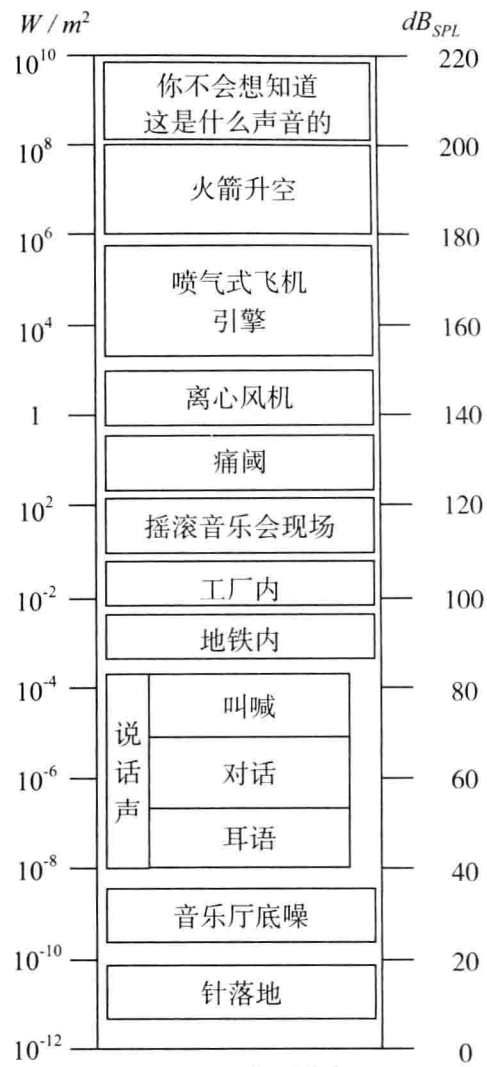


图 3.1 典型声级

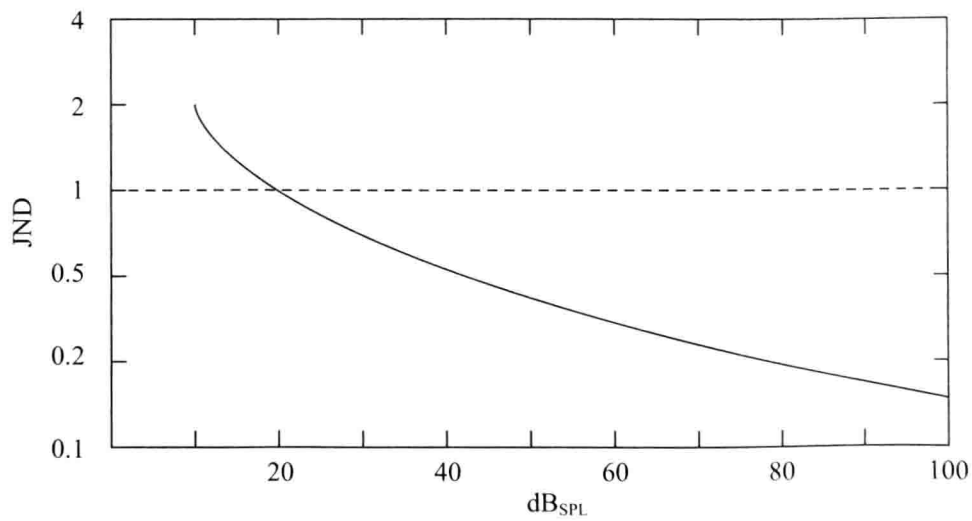


图 3.2 1 kHz 的最小可觉差别阈限 JND(Zwicker, Fastl 1999)

度。试着不改变振幅参数,用 MATLAB® 代码来播放 100Hz 正弦波和 1 kHz 正弦波,如将振幅设为 1.0,就可以得出听觉上的对比结果。

这种对声音信号强度听觉感知的偏差可用如图 3.3 所示的弗莱彻-曼森 (Fletcher,

Munson) 纯音等响曲线来解释。第一眼看上去会觉得曲线的形状有些奇怪,那么就让我们以 40 响度级曲线为例来说明弗莱彻-曼森曲线的意义。设有两个控制旋钮——一个用来控制纯音的频率,一个用来控制纯音的声压级(dB_{SPL})。如果保持声压级旋钮不变,从 20Hz 以线性步进向上扫频至 10,000Hz,在升至 3,000Hz 附近时我们都会感觉到响度变得越来越大,而接近 10,000Hz 时响度变大的趋势变得缓和——即使保持音量旋钮不变。为了感受到同样的响度,也就是说,要在 40 响度级这一曲线所覆盖的 20 至 10 kHz 频率范围内获得“等响度(equal loudness)”,就必须一开始在 20Hz 将声压级旋钮调至 80 dB_{SPL} ,在 1,000Hz 附近时降低至 40 dB_{SPL} ,然后在接近 10 kHz 时调至 50 dB_{SPL} 。这就是弗莱彻-曼森曲线所要说明的大致意思。响度级还有一个特殊的名称是方(phons,即它的单位)。也许你已经注意到,曲线的形状随方有着显著的变化。当方达到痛阈附近时曲线几乎变成了一条直线——如在高声压级的摇滚音乐会上会觉得所有频率的声音都是一样响的。也许这就是摇滚音乐会听起来很吵的原因之一吧。

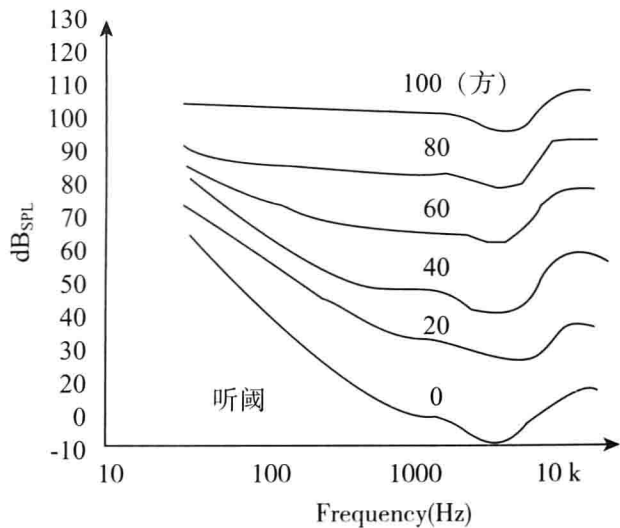


图 3.3 弗莱彻-曼森纯音等响曲线 (Fletcher, Munson 1933)

3.4 听觉掩蔽

假想你在一个安静的房间里以一个绝佳的声级在电视上收看费德勒与纳达尔的网球比赛,不知何故,你的室友突然想用真空吸尘器打扫房间,并且觉得这是一件非常重要的事情。这时来自电视的原本清晰、细致、声级绝佳的声音,就突然被这个用了 15 年的吸尘器的机械噪音所摧毁了。你开始听不到网球比赛的任何声音——因为它已经被吸尘器的噪音所掩盖了。这种一个声音被另一个声音所掩盖的情形就称为掩蔽(masking)。

掩蔽有两种基本情况:同时掩蔽(simultaneous masking)和分时掩蔽(non-simultaneous masking)。顾名思义,同时掩蔽是指两个声音事件,即掩蔽声(masker,上例中真空吸尘器的声音)和被掩蔽声(maskee,上例中网球比赛的声音)在同一时间发声时的掩蔽现象。也许你可以理解为,没有掩蔽声时,被掩蔽声的听觉阈限将更低,反之亦然。也就是说,当安静的房间里没有掩蔽声时,不用将电视音量提得很高,而当房间里有很多噪音时,只能通过提高电视的音量来提高信噪比(即 SNR,详见 5.1 节)。令人感兴趣的是,被掩蔽声的听觉阈限并不只是被掩蔽声和掩蔽声强度的函数,而与频率也有关系。图 3.4 显示了掩蔽阈限的变化规律 [盖尔芬德(Gelfand)修正, 2004],即保持掩蔽声频率为固定值 f_{masker} Hz 不变,被掩蔽声频率会随之在频率轴上变化(掩蔽声与被掩蔽声的强度均保持不变)。由图 3.4 可知,当被掩蔽声(上

例中的电视的声音)随掩蔽声频率 f_{masker} 变化时,其掩蔽效果会随着被掩蔽声频率与掩蔽声频率差的增大而减弱。当被掩蔽声频率等于 f_{masker} 时,掩蔽效果最明显——当吸尘器工作时,你也许真的要提高电视音量,尤其是当吸尘器噪音的频率与网球比赛的频率在相同的范围时。而如果吸尘器噪音频率(f_{masker})比电视的声音低很多,你就不必使用遥控器将电视音量提高很多了。比如吸尘器在隔壁房间工作,其高频声能量以及声音强度都会被隔墙减弱(隔墙在此可以看做一个滤波器,详见第7章),此时,从音质的角度看,收看电视将不会受到太大的影响。

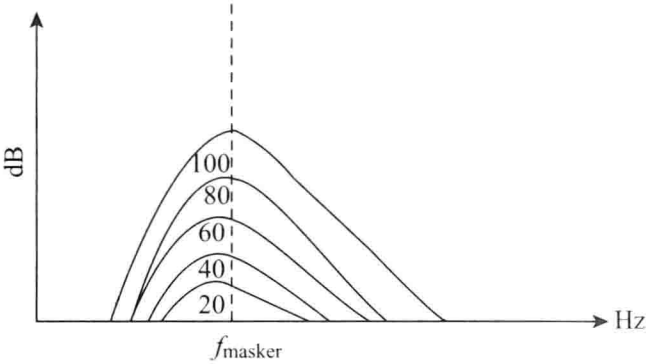


图 3.4 掩蔽阈限随频率变化的基本特征

与同时掩蔽不同,分时掩蔽或者称为时域掩蔽(temporal masking)是指掩蔽声与被掩蔽声非同步的、在不同时刻发声时的掩蔽现象。时域掩蔽又可分为后掩蔽(post-masking)和前掩蔽(pre-masking)。在下面这个情景中将会直观地感觉后掩蔽现象。当我们过于专注着自己的事情而穿行在首尔(Seoul)市中心的街头时,没有注意到人行道的灯已经变红了。这时坐在一辆大卡车驾驶室中的家伙疯狂地鸣着喇叭从你身边呼啸而过,幸好你在路旁边停住了,呼啸声也没损坏你的听力。虽然呼啸声只持续了1秒钟,而在接下来的200毫秒或更长时间内你可能将听不到任何声音。这就是后掩蔽现象。前掩蔽现象更为有趣,比如在上述过马路的情景中,所有的声音都同时出现在了后掩蔽阈中,即在汽车喇叭之后的声音都被掩蔽了。但事情还没有结束。有另外一些声音出现在掩蔽声之前,只要它出现的时间与掩蔽声音事件(汽车喇叭)足够接近,那么你可能听不到该声音。换句话说,先于掩蔽声的声音事件也有可能没有征兆地被掩盖——这就是前掩蔽。前掩蔽也是掩蔽声与被掩蔽声在强度和频率上的函数。上述三种掩蔽现象的时域特征如图3.5[茨威克(Zwicker)之后的结论,1999]所示。

很多音频压缩算法利用了人耳听觉的局限性和心理声学规律,其称为感知编码(perceptual codecs)。该类编码(包括编码和解码)中,MP3就是一个很好的例子,掩蔽特征在其中扮演了一个重要的角色,它可以在保证听觉舒适的情况下,降低一个给定音频信号的数据容量以利于提高网络中音频文件的下载和上传速度。

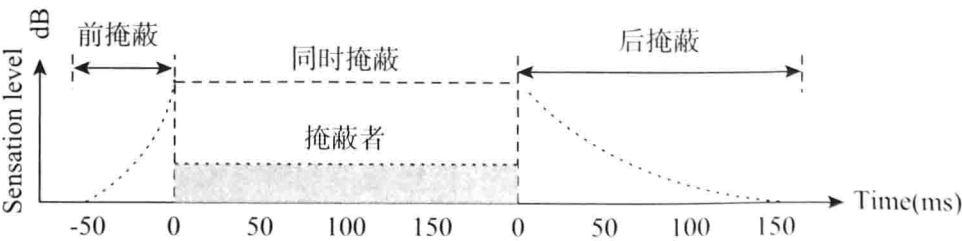


图 3.5 掩蔽特征曲线

4. 采样:离散化的艺术

到此为止,我们已经着重讨论了一些使模拟声音信号的采样以及数字化成为可能的人耳听觉系统的局限性。简单地说,采样就是通过被称为模拟数字转换器(ADCs, analog to digital converters)的设备将模拟信号转换成数字信号的过程,同时连续的模拟信号也就被编码成离散的数字信号,这一数字信号也可以称为原模拟信号的压缩版本。以电影为例——当我们在电影院里观看电影时,我们希望通过自己的视觉感知器官所看到的是运动物体、人物和其他事件的连续流畅的运动画面。但众所周知,这些电影画面实际上并非是完全连续的,它实际上是在一定的帧率设定下(每秒 24 帧、25 帧或 30 帧)由投影机投射到银幕上的画面。就像听觉系统一样,我们的眼睛只具备有限的采样容量或者说是采样率,当信息的采样率高于人眼时,人眼并不能感觉到。这就是为什么电影摄影机并不需要每秒 100 次的快门速度,因为我们在视觉感知上并不能区分每秒 24 帧、25 帧、30 帧和 100 帧采样率的差别。也就是说,每秒 50 帧或 100 帧,或者反过来为每帧 1/50 或 1/100 秒都有冗余并多少超出了感知范围。我们的感知器官和大脑对于历时性事件的捕捉和分辨能力是有限的。不过,如果降低电影的帧率,事情将完全不同,因为每秒的帧数越多,运动画面的渲染将会越平滑。

同理,对声音信号进行采样时,正如前文所述,我们最高只能听到大约 20 kHz 的声音,而对于强度也只能听到有限的动态范围。当我们说最高只能听到 20 kHz 的声音,则意味着当一个正弦波每秒振荡次数超过 20,000 周时,我们将感觉不到。至少从间断压缩或者持续的认知过程来看,不论这种感觉来自于机器还是自身大脑,人耳听觉的这种局限性都具有一定的优势。换句话说,每个采样点间的时间间距不需要达到无穷小(计算机系统中,无穷小会带来诸多问题),而事实上只需要足够小就能满足人耳听觉的需要,同时大脑也会认为这一经过采样后的信号(即数字化后的声音)与原模拟信号(非数字化的声音)是一致的。图 4.1 即能够说明上述的观点。

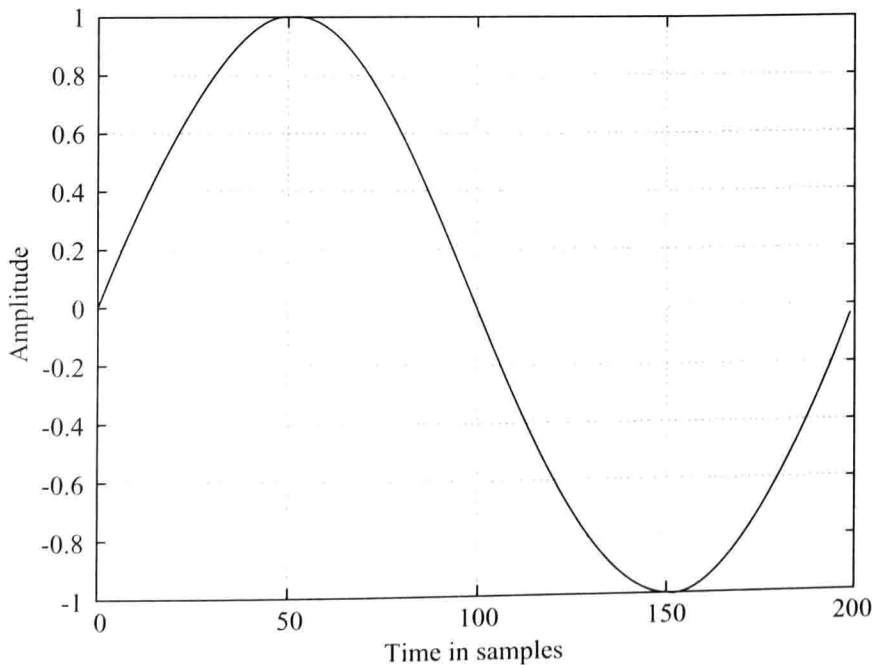


图 4.1 一个完整周期的 1Hz 正弦波形

尽管该图看起来非常平滑甚至是连续的,但事实并非如此。实际采样后的信号更像是图 4.2 所示(该图将图 4.1 放大至半个正弦波)。当仔细观察该图时,我们会发现它并不是一条平滑的曲线,而是沿正弦波形分布的一系列离散的点。就像电脑显示器上某种模拟的像素分辨率一样,当从足够远(无所谓足够远指的是多远,如像你去年在某个山顶上观看一样)的地方看一个数字画面时,画面是平滑而连续的,但将鼻尖凑近屏幕时,画面被放大了,你马上会看到一个以像素方式离散化后的人造画面。

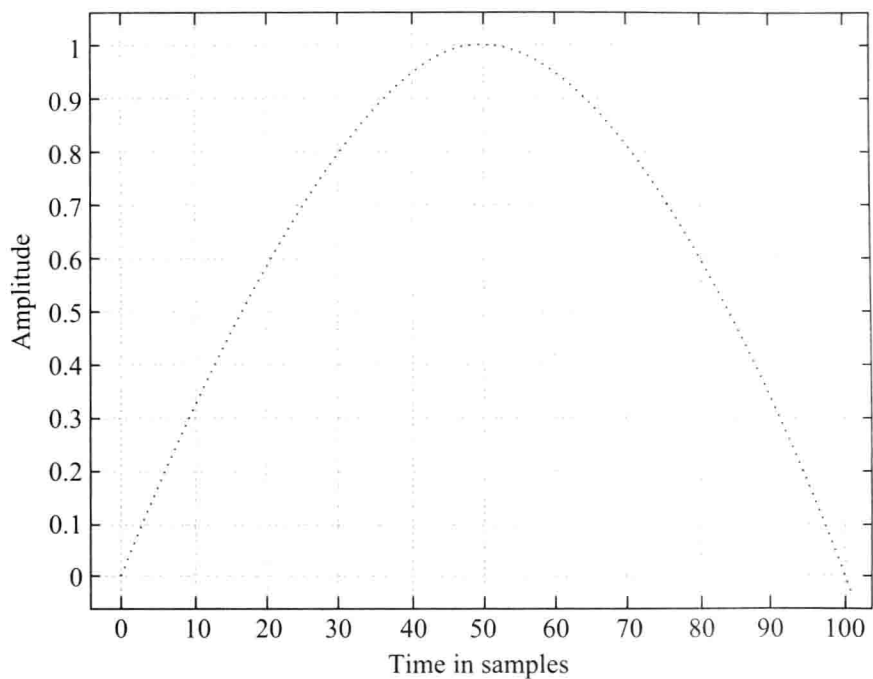


图 4.2 约半周期 1Hz 正弦波的采样信号,采样率 $f_s = 200\text{Hz}$

在此例中,我们采用 200 个离散的采样点来重构一个正弦波的完整周期,这 200 个采样点准确地对应着一秒钟的时长。因此一个正弦波在一秒钟完成一个完整的周期运动,那么以频率的定义,它的频率必然是 1Hz。对这个系统来说,采样率为 200Hz 或者每秒 200 个采样点,通常用符号 f_s 表示。由此也可以推断每个采样时间单元或者间隔,在此例中为 $1/f_s = 1/200$ 秒。这一时间间隔即指的是采样周期,通常用符号 T 表示。则采样率 f_s (Hz) 和 T (s) 有如下关系:

$$f_s = 1/T \tag{4.1}$$

或

$$T = 1/f_s \tag{4.2}$$

如图 4.3 为上述正弦波经采样后的四分之一周期波形,从图中可以看出相邻两个采样点之间的时间间距等于 T ,即 $1/200$ 秒。如图 4.4 所示为同一个正弦波采样后的半个周期波形,不过采样频率只有之前的四分之一,即 $f_s = 50\text{Hz}$ 。从中可以明显地看出重构该半周期的采样点数降低到了原来的 $1/4$ 。

在这一点上,我们也可以说时间是离散的——以 200Hz 的采样频率对信号进行离散化时,采样周期 $T = 1/200 = 0.005$ 秒或 5 毫秒(ms),也就是说在 2.5 毫秒或与 T 呈非整数倍的时刻,我们不能如下式一样获得一个具备有效值的采样点(设初始时间 $t = 0$)。

$$t = n \cdot T \tag{4.3}$$

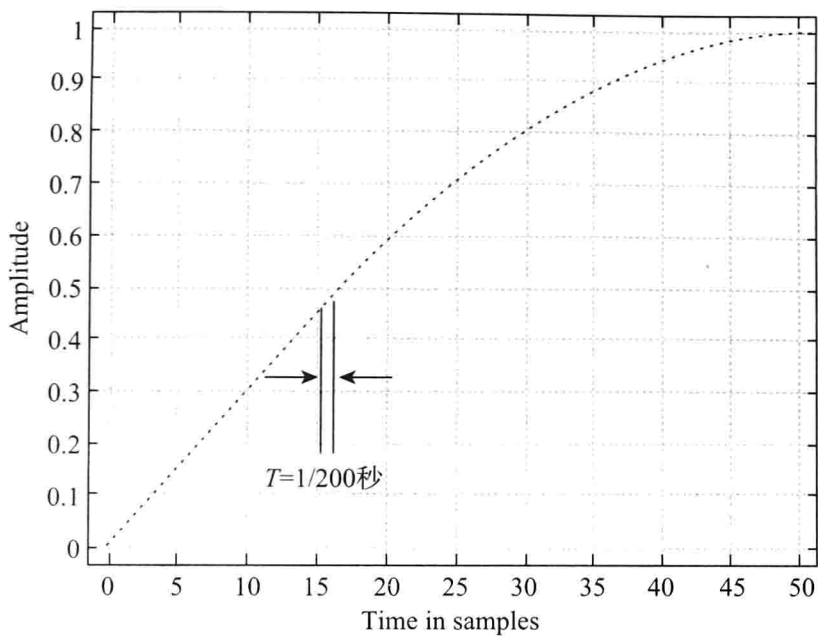


图 4.3 放大后的离散化正弦波形和采样周期 T

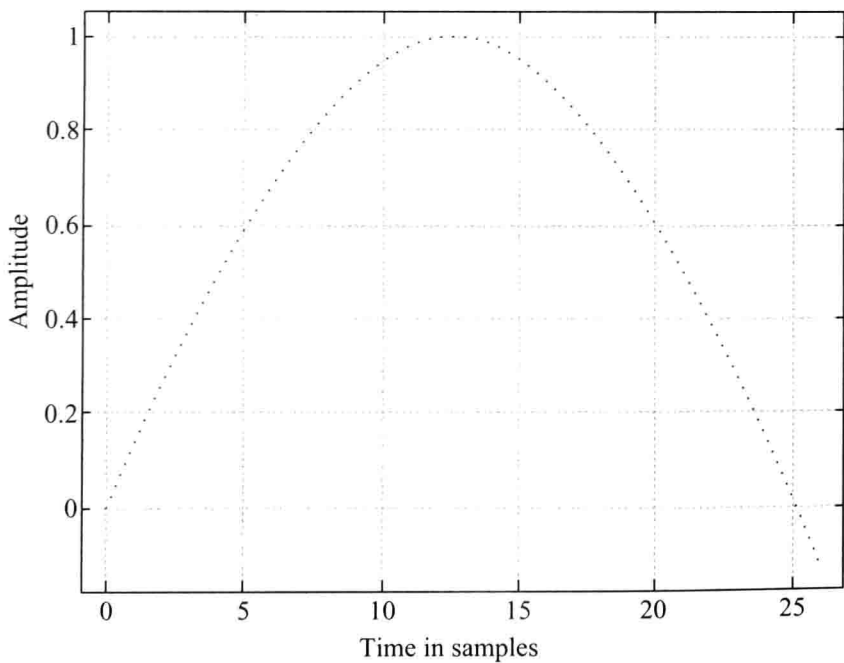


图 4.4 采样率 $f_s = 50\text{Hz}$, 1Hz 正弦波的半周期

式中 n 为整数,为采样点的时间序号,则在数字信号中,正弦波的每一个时间序号或采样点所对应的振幅可以用下式表示:

$$y(t) = \sin(2 \cdot \pi \cdot f \cdot t) \tag{4.4}$$

将 t 用式 4.3 代入式 4.4,则有:

$$y[n \cdot T] = \sin(2 \cdot \pi \cdot f \cdot n \cdot T) \tag{4.5}$$

又由于 n 是由整数表示的时间序号,例如图 4.5 中的第 0 个采样点,则有:

$$y[n \cdot T] = y[0 \cdot T] = y[0] = 0 \tag{4.6}$$

对于该正弦波的第 7 个采样点, $n=6$ (因为起始计数点是 $n=0$), 则有:

$$y[n \cdot T] = y[6 \cdot T] = 0.2 \quad (4.7)$$

对于模拟信号, 我们用圆括号“()”来表示, 对于数字信号用方括号“[]”来表示, 以清楚地区别连续信号和离散信号。知道了采样率 f_s 和采样周期 T 后, 就可以清楚地表示离散后的每一个采样点在时间上的位置了。通常会省略采样周期符号 T , 将 $y[n \cdot T]$ 记做 $y[n]$ 。回到本章一开始所介绍的 MATLAB® 代码, 就能清楚地了解时域连续的正弦波 $y(t)$ 是如何转变成时间离散的正弦波 $y[n \cdot T]$ 的了。也就是说:

$$y(t) = \sin(2 \cdot \pi i \cdot f \cdot t) \quad (4.8)$$

令 $t=n \cdot T$, 有

$$y|_{t=nT}(t) = y[nT] = \sin(2 \cdot \pi i \cdot f \cdot n \cdot T) \quad (4.9)$$

因 $T=1/f_s$, 则 MATLAB® 代码可以写作:

$$y[n] = \sin(2 \cdot \pi i \cdot f \cdot n \cdot T) = \sin\left(2 \cdot \pi i \cdot f \cdot \frac{n}{f_s}\right) \quad (4.10)$$

由于 n 是与采样点时间序号对应的整数, 设 $n=[0:44099]$ 可得到时长为 1 秒的正弦波 ($[0:44099]$ 表示一个步进为 1 的值为 0 至 44099 的整数数列)。

$$y[0:44099] = \sin\left(2 \cdot \pi i \cdot \frac{f}{f_s} \cdot [0:44099]\right) \quad (4.11)$$

令 $f=1$, 能够得到采样率 $f_s=44,100\text{Hz}$, 时长为 1 秒, 频率为 1Hz 的信号。

4.1 采样定理

在正弦波的 MATLAB® 代码中, 我们采用 44,100Hz 的采样率, 即用每秒 44,100 个采样点来重构正弦波。那么采用其他的采样率, 或者已经采用了不同的采样率呢? 答案取决于所需要重构的正弦波的频率, 特别是所要重构的最高的正弦波频率, 即采样率的大小必须满足模拟信号 (此例中是单正弦波) 重构后不会产生能听得到的痕迹或失真。正如前面所说的, 人耳最高能听到 20,000Hz 的声音, 那么假设的最佳采样率 $f_s=20,000\text{Hz}$, 由于 $f=1/t$, 则等同于采样周期为 $1/20000$ 秒, 或者反过来说每秒 20,000 个采样点。尽管这可能是一个合理的推测, 但实际却需要该采样率的两倍。更准确地说, 采样率至少应该是声音信号的最高频率分量的两倍, 即众所周知的采样定理 (sampling theorem), 其中 $f_s/2$ 指的是奈奎斯特频率 (Nyquist frequency), 以其发现者哈里·奈奎斯特 (Harry Nyquist) 命名。采样定理可用以下式 4.12 和 4.13 表示:

$$f_{\max} < \frac{f_s}{2} \quad (4.12)$$

或

$$f_s > 2 \cdot f_{\max} \quad (4.13)$$

以一个 1Hz 的正弦波为例, 由于只包含一个频率分量, 这意味着我们只需要高于 2Hz 的采样率, 或者每秒只需要 3 个采样点。在 MATLAB® 代码中, 生成一个 440Hz 的正弦波, 最低采样

率要高于 $440 \cdot 2 = 880\text{Hz}$ ——对于任何频率,每秒最少需要超过 880 个采样点。因此,人耳听觉的频率上限在 20,000Hz 左右,由采样定理可知,我们需要高于 40,000Hz 的采样率。CD 唱片即采用每通道量化深度为 16 bits、采样率为 44.1 kHz 的标准(一张 CD 有两个通道)。那为什么采样率是 44.1 kHz 呢?城市中的传说大致是这样的:回到 20 世纪 80 年代中期,两大工业巨头索尼(Sony)和飞利浦(Philips,其他合作者还包括 CBS/Sony 和 Polygram)致力于发展音频 CD 标准,采样率选择 36 kHz 还是 44.1 kHz 成为了问题。经过对这两种采样率标准的反复争论,最后由于以下原因选择了 44.1 kHz 作为标准:

人耳听觉频率上限是 20,000Hz

采样定理要求采样率 $f_s > 40,000\text{Hz}$

需要有额外的频率余量来支持 40 kHz 的采样频率

一张 CD 必须能容纳一首《贝多芬第九交响曲》

有趣的是,现行 CD 技术参数的标准之一是由于索尼和飞利浦决定以《贝多芬第九交响曲》的长度作为 CD 播放长度的基准。这样,经过一些研究之后,他们发现这部交响曲在演奏到《欢乐颂》乐章时大都在第 66 至 74 分钟之间。具有讽刺意味的是,按他们的说法,在历史上第一张 CBS/Sony 制作的 CD 唱片标题并不是《贝多芬第九交响曲》,而是比利·乔的《第 52 街》(52nd Street),以至于其他,正如人们所说,这是一段历史。而现在,我们常以 24 bits 作为振幅量化深度的标准,采样率为 96 kHz 甚至高至 192 kHz。是否有必要选择如此之高的采样率标准目前仍存在争论,很多争论似乎在于支持或反对这种必要性。鉴于至少需要整章或两章的篇幅来解释其中的技术和非技术因素,这部分内容不在本章讨论。

4.2 频率混叠

也许到这里,你会问:如果采样率小于音频信号最高频率分量的两倍时,也就是说,当 $f_s/2 < f_{\max}$ 时,会出现什么情况呢?答案是会导致一种被称为“频率混叠”(aliasing)的特殊数字失真。有趣的是,这种数字失真会出现原信号中所没有的频率分量。这种失真由于“减采样”(under-sampling)造成的,有时也许还会出现对作品有所帮助的奇妙声音,不过在数字域通过采样对模拟音频信号进行精确重构而言,这种失真是不需要的。在介绍完频域特征及傅立叶变换后,第 8 章会详细讨论频率混叠是如何产生的。而现在,本节从直观的时域来领会其基本思想。

仍以图 4.5——即采样率 $f_s = 100\text{Hz}$ (每秒 100 个采样点)的 1Hz 正弦波为例。1Hz 正弦波,采样率 $f_s = 100\text{Hz}$ 符合采样定理的标准,奈奎斯特限制频率位于 50Hz——任何低于 50Hz 的正弦波都可以在数字域清楚地重构。因此,此例中($f = 1\text{Hz}$)没有出现频率混叠失真。由图可知,1Hz 正弦波在 1 秒内完成一个周期,采样周期 $T = 1/100$ 秒,则重构该 1Hz 正弦波所需要的采样点数为 100。

现在将正弦波频率提高至 5Hz,保持其他参数不变——采样后的结果如图 4.6。再一次确认采样周期 T 和采样点数目保持不变,仍然是 1/100 秒和 100 个采样点,这样 5Hz 的正弦波在一秒内有 5 个完整周期而不是 1 个。之前我们用所有 100 个采样点来重构一秒内只有一个周期的 1Hz 正弦波,而现在只能用 $100/5 = 20$ 个采样点来重构一个 5Hz 的周期。

进一步将正弦波频率提高至 20Hz,预期的结果如图 4.7。我们再次观察 20Hz 正弦波中一个由采样点重构的周期——只需要 5 个采样点就可以重构一个 20Hz 正弦纯音的完整周期。虽

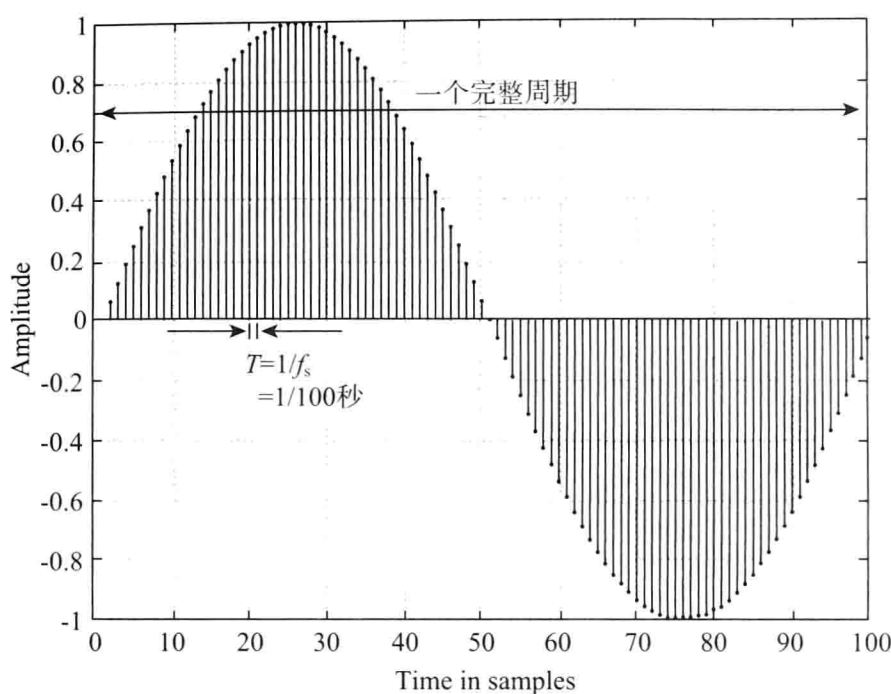


图 4.5 采样率 $f_s = 100\text{Hz}$, $f = 1\text{Hz}$ 的正弦波

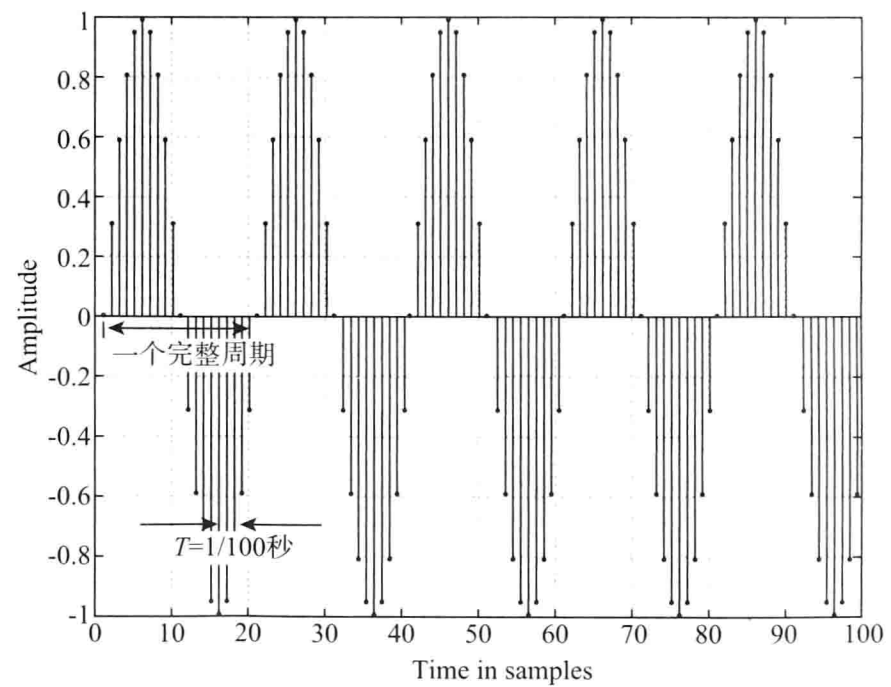


图 4.6 采样率 $f_s = 100\text{Hz}$, $f = 5\text{Hz}$ 的正弦波

然其波形和听感仍然像一个正弦波(试着在 MATLAB® 代码中改变采样率和频率参数),但已经损失了正弦波的外形特征。

现在把振幅包络进一步挤压,并把频率提高至奈奎斯特限制频率 50Hz,事实上,此时正弦波已接近图 4.8 所示。由图 4.8 可知,恰好低于 50Hz 的正弦波被压缩成一系列振幅为 +1.0 和 -1.0 的间断性采样点,每一对采样点能够在极值处重构频率接近 50Hz 的正弦波的完整周期特征。我想这已经能够看出是一个周期,并且其图形形态完备,也就是说,我认为这仍然能称为正弦波的最小特征(再一次声明,这不是真正可行的做法,不过为了看看最后会发生什么情况,让我们试着这样做)。

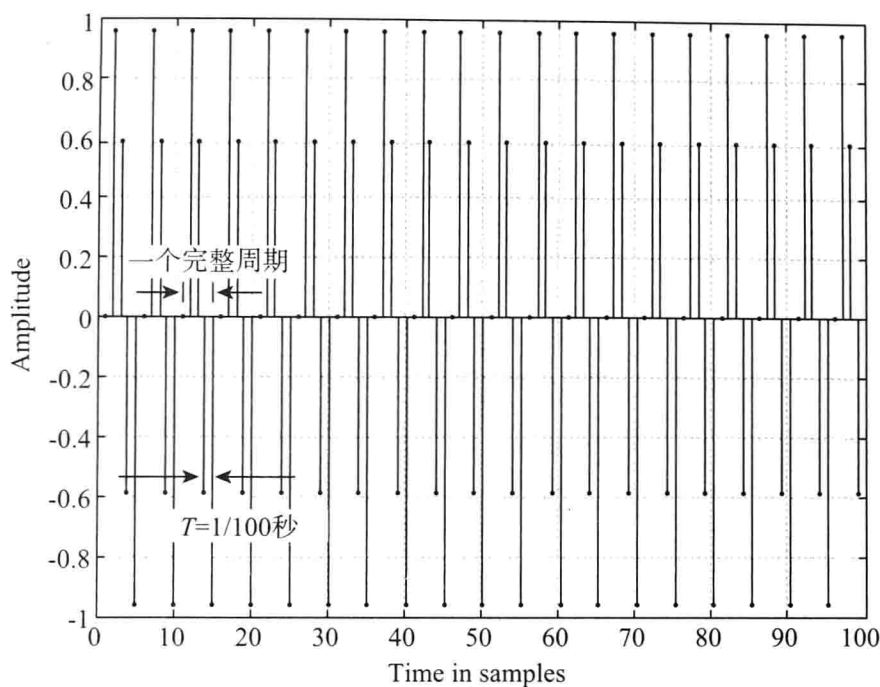


图 4.7 采样率 $f_s=100\text{Hz}$, $f=20\text{Hz}$ 的正弦波

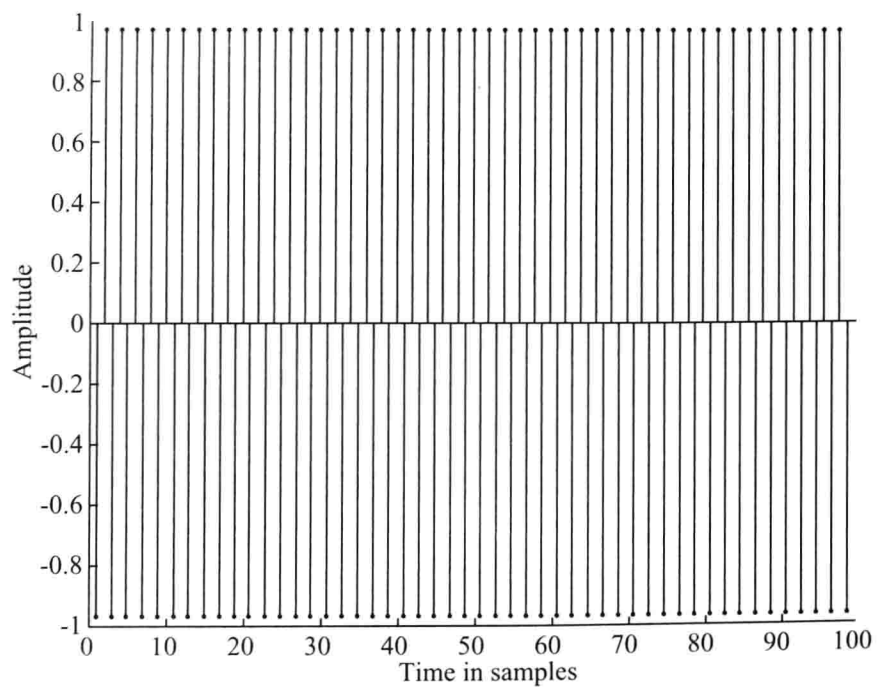


图 4.8 采样率 $f_s=100\text{Hz}$, 频率 f 接近 50Hz 的正弦波

至此,我们已经将 100Hz 采样率重构正弦波的能力扩展到了极限,任何更高频率的正弦波都不可能在该采样率下重构。在 50Hz 附近,用两个采样点就能在最小限度上重构一个正弦波,但对于任何更高的频率意味着每个完整的正弦周期只能用一个或更少的采样点来重构(即随着频率的增加,采样点数将呈减少的趋势)。只用一个采样点是不可能描述一个正弦波的(当然也不可能采用 0 个采样点!),因为至少需要两个位于相反峰值上的采样点才能描述一个正弦波的特征。如果使用更高的采样率,如 200Hz,就可以轻松地对 50Hz 或 60Hz 的正弦波进行采样,因为在该采样率下每秒钟拥有更多的采样点数来进行重构。但对于 100Hz 的采样率,我们

的讨论现在已经到了终点。这是一个产生频率混叠失真的临界点。为了解释这种频率混叠失真,让我们由 50Hz 开始让正弦波的频率超出奈奎斯特限制频率,看看真正得出的图像会产生什么样的结果,如图 4.9。

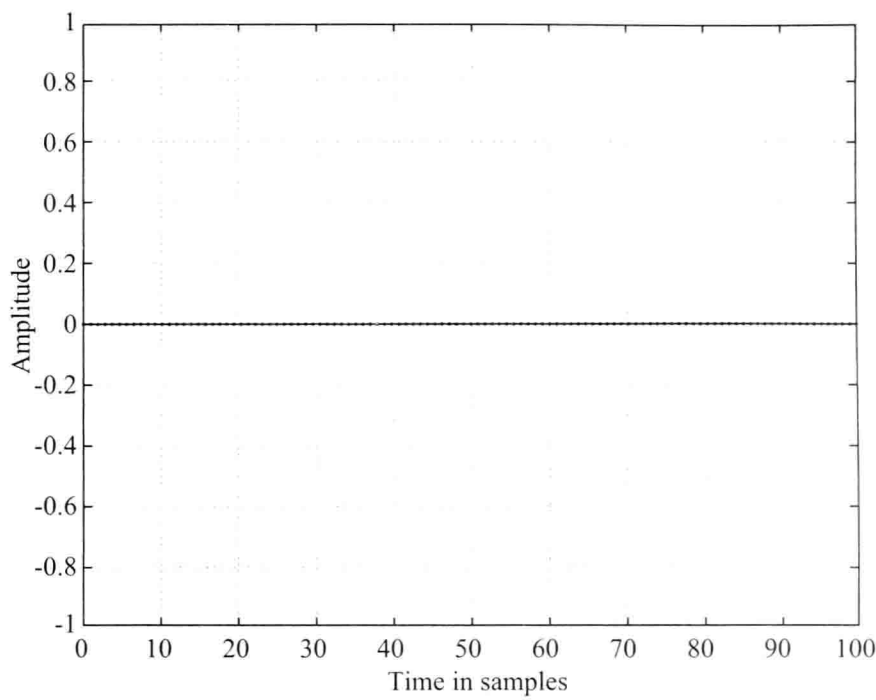


图 4.9 采样率 $f_s = 100\text{Hz}$, $f = 50\text{Hz}$ 的正弦波形

发生了什么情况? 由图 4.9 可知, 50Hz 正弦波经采样后完全等同了一个 0Hz 的正弦波, 而 95Hz 正弦波 (图 4.10) 变成了一个反转 (相位倒转 180°) 的 5Hz 正弦波, 而 105Hz 正弦波 (图 4.11) 与 5Hz 正弦波 (非反转) 完全相同。如果忽略上述图形中正弦波相位倒转的情况, 这些高于奈奎斯特频率的正弦波被频率混叠为更低频率的正弦波——这种失真就是一种被称为减采样的过程, 指的也就是频率混叠。此例中, 95Hz 正弦波被频率混叠为 5Hz 正弦波 (相位倒转 180° 或 π), 而 105Hz 正弦波则被伪装成了 5Hz 正弦波 (无反转)。人耳听觉系统的一个有趣特征是, 在聆听纯音时, 人耳并不能分辨相位差, 也就是对我们而言, 5Hz、减采样后的 95Hz 和 105Hz 尽管相位不同但实际上听起来是一样的 (如果人耳能听到 5Hz 声音的话)。如果一个信号包含多个相位不同的正弦波, 那么由于相长及相消干涉作用, 对这种复合音的听觉感知将完全不同 (相长及相消干涉作用将在第 4 章第 4 节讨论)。尝试在 MATLAB[®] 代码中改变频率参数来确认一下频率超过奈奎斯特限制频率时, 所发生的低频频率混叠现象。上述利用每周期的采样点数对频率混叠的阐释是频率混叠的直观表现, 本书将在第 6 章和第 8 章开发和利用更强大的信号处理工具对频率混叠做进一步论述。图 4.12 给出了频率混叠理念的图解。

综上所述, 频率高于奈奎斯特限制频率时必然会出现低频频率混叠。这种现象的有趣之处在于, 音乐信号很少只由一个正弦纯音组成, 但也不是无穷多个正弦纯音的组合。当采样率选择不当时, 可以想象所导致的频率混叠累积现象是多么严重。

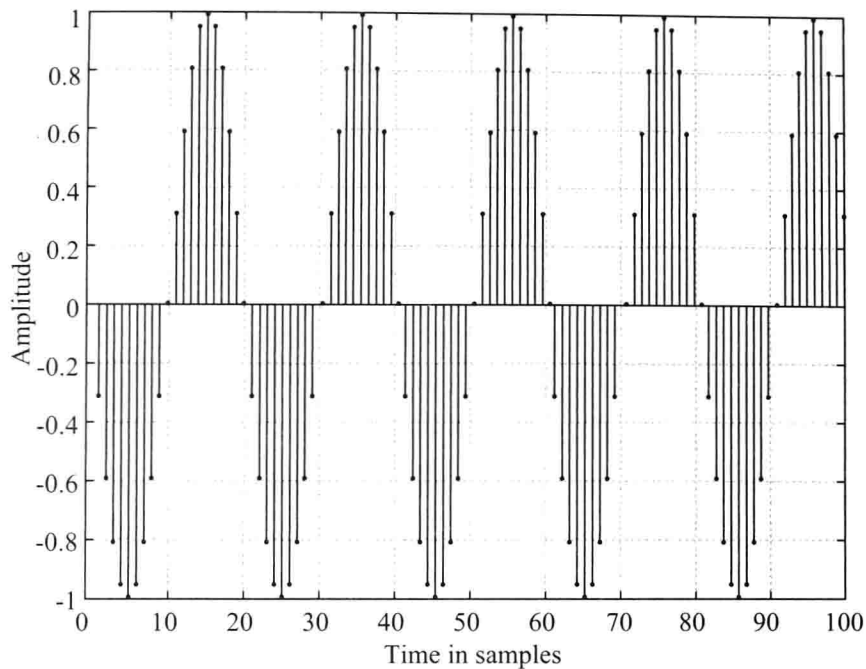


图 4.10 采样率 $f_s = 100\text{Hz}$, $f = 95\text{Hz}$ 的正弦波形

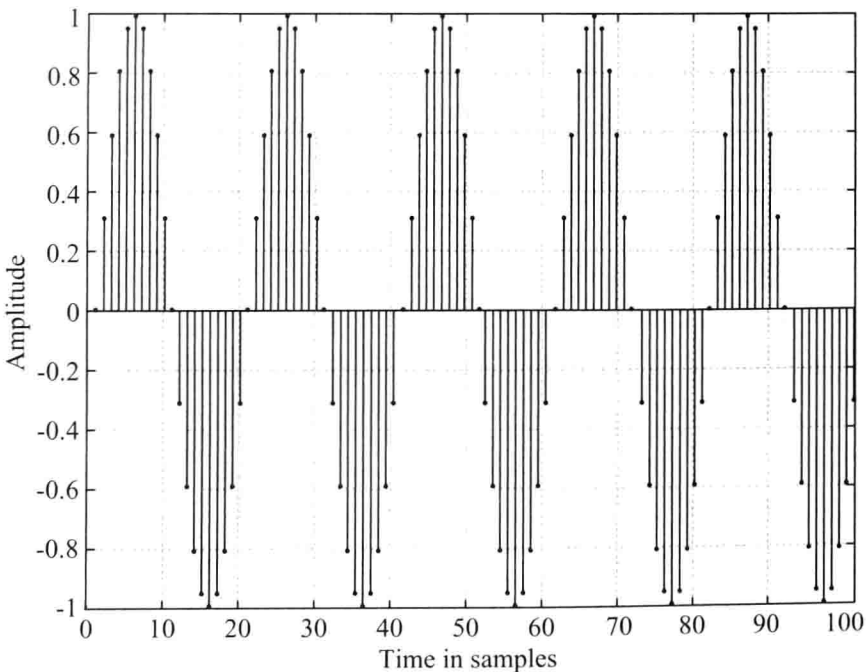


图 4.11 采样率 $f_s = 100\text{Hz}$, $f = 105\text{Hz}$ 的正弦波形

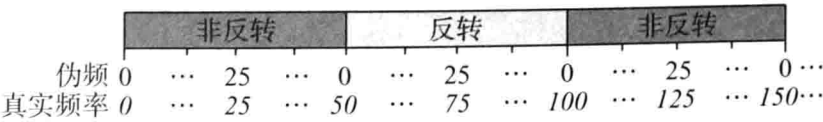


图 4.12 采样率 $f_s = 100\text{Hz}$ 时的频率混叠模式

5. 量化与脉冲编码调制(PCM)

至此,我们的讨论更多关注于声音的频率分量,而没有注意到在模拟信号数字化过程中每

个采样点上幅值的变化及其相关问题。这一节关注的问题是在数字域重构原模拟信号时对声音振幅的劣化处理。采样后的模拟信号振幅的质量、分辨率和精度取决于影响量化误差的比特深度(bit depth)。量化误差(ε)是指数字离散化后的幅值与模拟信号幅值的差,如式 5.1 所示,其中 n 为采样点序号。

$$\varepsilon = x(t) \big|_{t=nT} - x[n \cdot T] \tag{5.1}$$

如前所述,音频 CD 包含两通道的音频信息,每通道采样率为 44.1 kHz,量化深度为 16 bits,相当于 65,536 个离散的幅值。例如,将最小和最大幅值归一化至 -1.0 和 +1.0 之间、一个 16 位的数字系统则意味着将 ± 1.0 之间分成 65,536 个离散的单元。也就是说,如果有某个值,假设处于 0 到 1 之间,如 0.4,则对于有 65,536 点的整数系统,四舍五入取整(rounding,将数加上 0.5 再去掉小数点后的尾数)、去尾取整(flooring,直接去掉尾数)和进一取整(ceiling,使用下一个大于该数的整数)都是量化时常用的方法。以相同的步进划分振幅的方法称为均匀量化(uniform quantization)。对幅值进行量化,并且使用特定的采样率来存储或记录数据称为脉冲编码调制(PCM,pulse code modulation)。也许这个术语让人有些疑惑,因为除了分析数字化后幅值的二进制编码结构外,PCM 系统本身并没有脉冲响应。如在 16 位二进制格式中,数 2,175 的二进制码是 0000 1000 1000,1001,如图 5.1 所示。

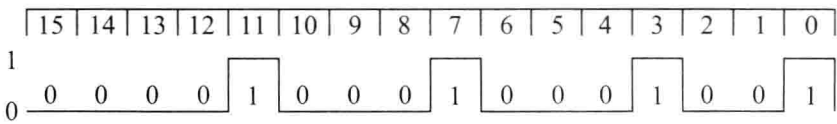


图 5.1 PCM 思想示意:16 位二进制格式中数 2,175 的表示方式

由图 5.2 可知,时间离散与幅值离散的正弦波重构一个原正弦波的粗劣版本。一个数字系统的比特分辨率越高, y 轴上的振幅刻度就被划分得越细,就像提高采样率 f_s 可以减小采样时间间隔(T)一样。图 5.3 给出一个保持采样间隔不变,比特分辨率的降低导致量化误差变大的例

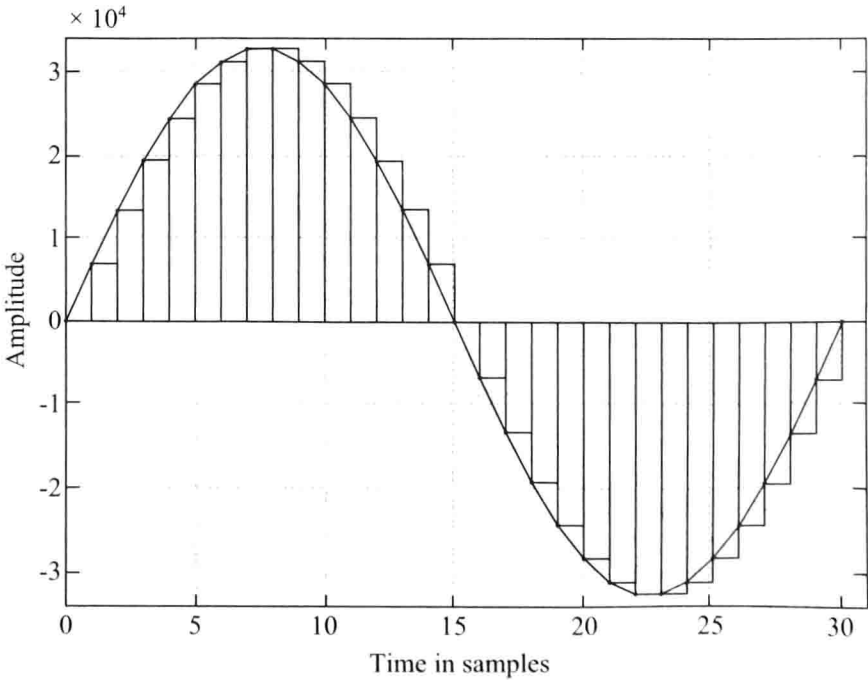


图 5.2 幅值的量化

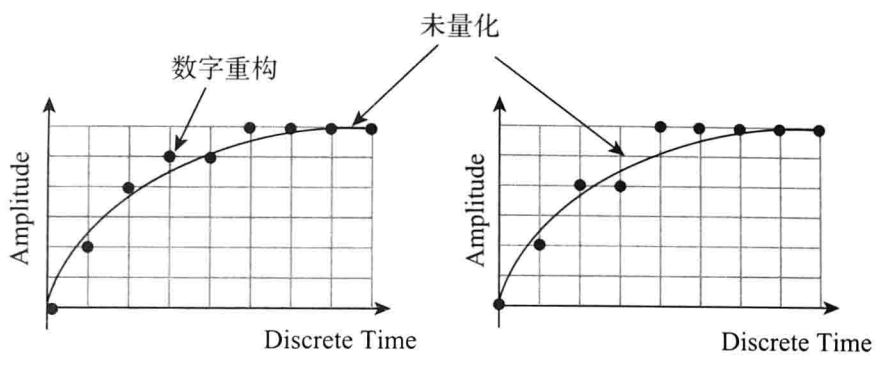


图 5.3 比特分辨率及其对量化的影响

(左图为较好的量化分辨率,右图为较差的量化分辨率)

子——左图所示为较高的量化分辨率,而右图中较低的量化分辨率限制重构 y 轴振幅所用的点数。

信噪比 (SNR) 和量化信噪比 (QSNR)

众所周知,对音乐声或语言声来说,信噪比 SNR (singal-to-noise ratio) 是一个量度有用信号强度的标准方法。接下来将简要讨论与背景噪声或本底噪声有关,以及与比特分辨率和量化信噪比 QSNR (quantization singal-to-noise ratio) 有密切关系的问题。信噪比 SNR 定义为信号与噪声的能量比,通常用式 5.2 来计算平均信噪比。

$$SNR = 20 \cdot \log_{10} \left(\frac{A_{signal}}{A_{noise}} \right) \tag{5.2}$$

高信噪比的系统往往如所期待的那样,信号大于噪声,或者反过来,噪声低于信号。拥有一个良好的信噪比对演奏者来说,如电吉他手,就意味着吉他音量旋钮会拧到最高位置,以掩盖在录音棚里用来录制吉他的录音系统中来自于吉他本身、功率放大器、传声器以及其他硬件设备或录音空间的本底噪声。

我们在对模拟信号进行量化时,由于一个数字系统不可能拥有无穷大的比特分辨率,无可避免地会引入一些如式 5.1 所示的量化误差。问题在于如何确定既定量化深度下的所谓的量化信噪比或 QSNR 呢? QSNR 的定义如下,其中 N 为比特分辨率:

$$QSNR = N \cdot 6.02 + 1.76dB \tag{5.3}$$

QSNR 的近似值来自于有用信号的最大幅值 (2^{N-1}) 与模拟和数字幅值的最大误差 1/2 bit 的差值。1/2 bit 的误差是指模拟信号幅值在两个相继整数值之间的选择误差,如对于值 60.5 可量化为 60 和 61 所带来的误差。用 2^{N-1} 而不是 2^N 的原因也很简单。回到图 5.2,一个 16 bit 的正弦波意味着采用 0 至 65,536 之间的所有可能的整数。如果我们想得到关于 x=0 轴(用符号表示)的振动,则整数位于 -32768 至 +32767 之间(注意,如果包括 0 在内,所有可用的整数正好是 65 536 个)。因此,如对于一个 16 位信号,最大幅值的绝对值 [如 $abs(-1) = 1$] 即为 $2^{16} / 2 = 2^{15} = 32768$ 。则有:

$$QSNR = 20 \cdot \log_{10} \left(\frac{A_{signal_max}}{A_{noise_max}} \right) = 20 \cdot \log_{10} \left(\frac{2^{N-1}}{1/2} \right)$$

$$\begin{aligned} &= 20 \cdot \log_{10}(2^{N-1}) - 20 \cdot \log_{10}(1/2) \\ &= 20(N - 1) \cdot \log_{10}(2) + 6.02 \\ &= 20 \cdot N \cdot \log_{10}(2) - 20 \cdot \log_{10}(2) + 6.02 \\ &= 6.02 \cdot N \end{aligned}$$

(5.4)

这一来自于 QSNR 的重要结果可以用来大致估计数字系统的动态范围——即 QSNR 基本上告诉我们,每增加 1 bit 动态将增加约 6 dB。因此,对于一个 16 位数字系统,如 CD 音质的音频,我们所期望的动态范围为 $16 \cdot 6 = 96 \text{ dB}$ 。

6. DC 直流分量

“DC”一词实际上来自于电气工程协会使用的“直流电”(direct current)一词。在数字信号处理中,“DC”一词指不随时间改变的固定偏移量。如输入信号 x 和对应的输出信号 y ,偏移量或者说 DC 分量如式 6.1 所示,其中 C 为常数:

$$y(t) = x(t) + C$$

(6.1)

以正弦波为例,偏移量为:

$$y(t) = A \cdot \sin(2 \cdot \pi \cdot f \cdot t) + C$$

(6.2)

在不改变其波形的情况下,可根据上式将整个正弦波在 y 轴上垂直平移。这样,在理想情况下,人耳听觉系统并不能分辨出一个信号是否包含 DC 分量。不过相对于模拟系统,在数字系统中,DC 分量会导致一些潜在而严重的失真问题。原因是,对于数字系统,只能用有限的比特数来存储和重构数据或波形,如音频 CD 是 16 bit。又如在图 5.2 的正弦纯音中加入一个值为 10,000 的 DC 常量,则波形如图 6.1 所示。

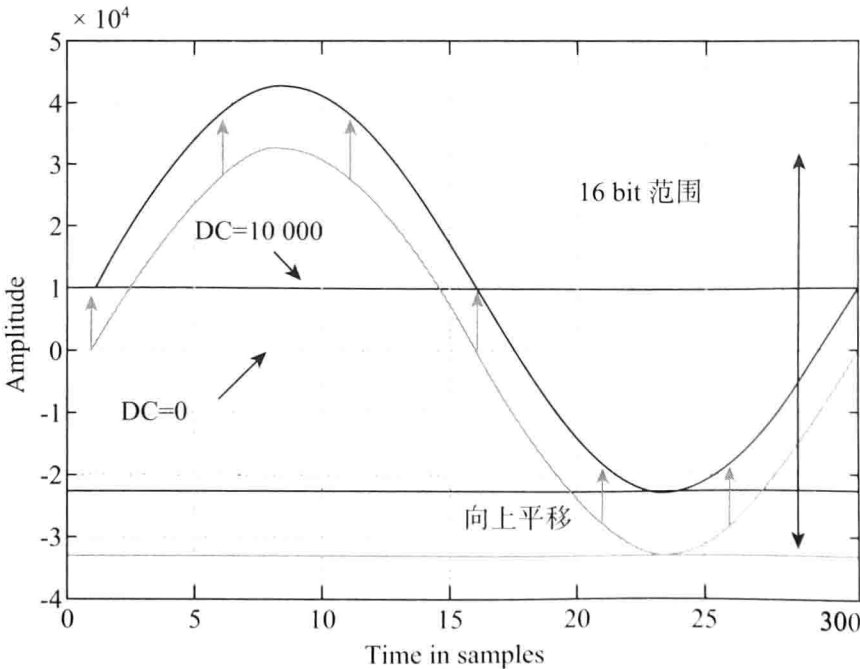


图 6.1 正弦波的 DC 偏移

注意到以 16 bit 符号化的整数系统对该波形进行存储和重构时,在波形的顶部会产生削波 (clipping, 相关专题见下节。而在 MATLAB® 中,声音文件的幅值将会归一化至 -1.0 和 +1.0 之间)。当 DC 偏移量产生削波时,波形以及该波形对应的声音必然会发生失真和变化。因此,在对信号进行采样之前,应当关注如何消除 DC 偏移量。我们稍后将介绍通过高通滤波器 (high-pass filter) 来消除 DC 偏移量的方法。而一种直观的、不用滤波器 (高通滤波器将在第 7 章讨论) 来消除 DC 偏移量的方法是计算波形的幅值算术平均值 (指幅值振荡所在的 y 轴上的算术平均值),然后在原波形中移除该平均值,如式 6.3。

$$y(t) = x(t) - \bar{x} \quad (6.3)$$

在 MATLAB® 中可以直接调用如下内部均值函数:

```
y = x - mean(x)
```

代码 6.1

这一均值当然就是算术平均值——即对向量 x 中的所有元素数值进行求和再除以向量的长度 (向量中元素的个数)。

7. 失真与方波

上一节简要介绍了削波的概念。当信号过强或存在 DC 分量而导致信号削减或截止时会产生削波,在这种情况下需要对原信号进行适当的调整。这些调整包括在进行某些信号处理 (或采样) 前减弱信号本身的强度,或如上述所说去除 DC 偏移量。某些情况下,削波实际上是故意制造的效果——如在音乐中经常利用削波来改变音源的音色。电吉他手也许会很熟悉削波这一术语,并且通过踩箱、踏板或功率放大器来制造削波,从而制造出接近摇滚吉他的音色。上述削波、失真或过载效果有时也可以简单地用下式表示:

$$y(t) = a \cdot x(t) \quad (7.1)$$

参数 a 是一个能对输入信号 x 进行缩放的增益因子。以下伪代码能产生简单的失真效果:

```
if (| a*x | > clippingLevel)
    y = sign(x)*clippingLevel;
else
    y = a*x;
end
```

如果输入信号是一个正弦波,逐步增大该增益,所观察到的特征如图 7.1 所示 (设超过 16 bit 限制的幅值会产生削波),其中缩放因子 a 都由 1.0 开始,每一个图形逐一倍增。

除了该正弦波逐步变成了一个方波 (square wave) 之外,似乎该图没有什么特别之处。一般来说,纯音 (或许听起来很单调) 是一个单一频率的正弦音。它有着明确无误而独特的音质和音色,已经在早期的电子音乐中使用,特别是在德国,一些作曲家如卡恩海因茨·施托克豪森 (Karlheinz Stockhausen) 乐于用振荡发生器来进行实验,但它并没有太奇妙的音乐性。随着正弦波越来越接近方波,情况也就变得越来越有趣和复杂。那么,为什么一个类似方波的形状会比

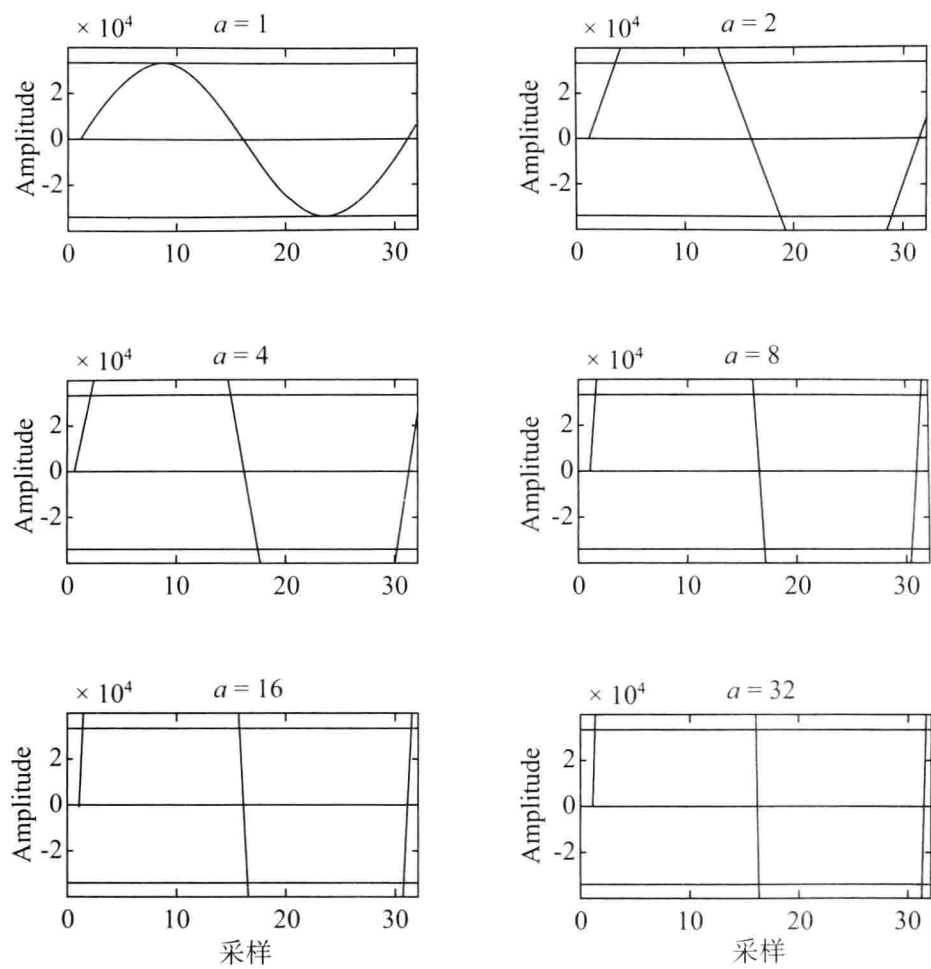


图 7.1 削波与失真, 缩放因子 a 由 1 倍增至 32 倍

较令人感兴趣呢？在没有介绍傅立叶变换和信号的正弦分解之前,很难给出恰当的解释。但从声音事件的复杂性及丰满度(丰满度最低的是正弦波,丰满度最高的是白噪声)的一般规律来看,全时域内的振幅变化越剧烈,则声音的丰满度越高。正弦波非常平滑,采样点与采样点之间变化缓慢,而较复杂的波形,如方波,采样点与采样点之间变化剧烈,而噪声信号就是一个极端的例子。如图 7.1 所示,正弦波慢慢地变为方波,显著的特征是波形边缘变得越来越陡峭,从而产生了一个更丰满、更复杂,也许也更令人感兴趣的声音。

回到之前吉他的例子,图 7.1 中的后几个波形虽然不是正弦波,但对于吉他的声音,近似失真效果会使原吉他声音变得更丰满,在某种程度上也更为有趣。也就是说,削波失真效果在处理后的信号中加入了奇次谐波(odd harmonics,原因将在第 8 章讨论)。因此,削波失真又称为“谐波失真”(harmonic distortion),因为它在原波形中加入了奇次谐波,同时通过波形调制,使音色变得更丰满,或者至少可以说能够产生“奇怪的谐和感”。

抖动

抖动(dithering)是指在量化和采样之前,在模拟输入信号中加极少量[次比特(sub bit resolution),即低于一个比特]的伪随机噪声(error ϵ),如式 7.3。这种方法听起来有点另类,因为在信号中加入噪声看上去不可思议,而我们希望信号尽可能的干净。

$$x(t)_{dithered} = x(t)_{original} + \epsilon$$

(7.3)

当你用耳机听 CD 唱片时,如果你在喜欢的曲目的渐弱部分提高音量,有没有发现一些隐藏信息或者类似的东西,会听到一些令人厌烦的、嗡嗡的、模糊的、砂粒状的声音。这实际上就是粒化噪声。这是在采样和量化时人为加入的,通常情况下都不会很明显,因为通常不会播放唱片尾部 1 或 2 bit 强度的延长段落。因其声级很低,一般不容易觉察,听者根本不会认为这种人工失真是个问题。但对于那些从事声音分析、设计、编辑和开发 DSP 算法或者音乐创作的作曲家或声音研究者来说,这些粒状噪声一部分来自于削波(希望在商业 CD 唱片中这种问题不是有意造成的)。抖动能缓解与上述量化所产生的相关问题。要理解抖动的作用,让我们看看四舍五入法量化的过程。图 7.2 的上图为一个指数衰减的余弦波和以 4 bit 深度的水平量化网格线重构的量化波形。

如图 7.2 的中图所示,由于较低的信误比(signal to error ratio)QSNR 衰减得很快,这意味着原信号强度与量化误差的能量比越来越小。这就是某些潜在问题的根源——在第 20 至第 110 个采样点之间,量化后的波形模式化或规则化了,并具备了方波的类似特征(极端的例子稍后介绍),无法对原信号进行重构。此外,由于新加入的谐波失真会产生频率混叠现象——在信号中加入的奇次谐波越多,产生高于奈奎斯特限制的频率分量的可能性就越大(详细内容见第 8 章)。现在的问题在于,类似方波出现的重复模式化现象加入了不需要的谐波,尤其是原信号中没有的奇次谐波。解决这一问题的办法当然是提高比特率,用更高的比特深度来对低幅值区域细微的变化(低 QSNR)进行重构,从而增加每提高 1 比特量化深度所带来的 6 dB 的动态范围。但这种做法的代价很高,并且无法预知需要处理的是什么信号,它的响度如何,以及它何时

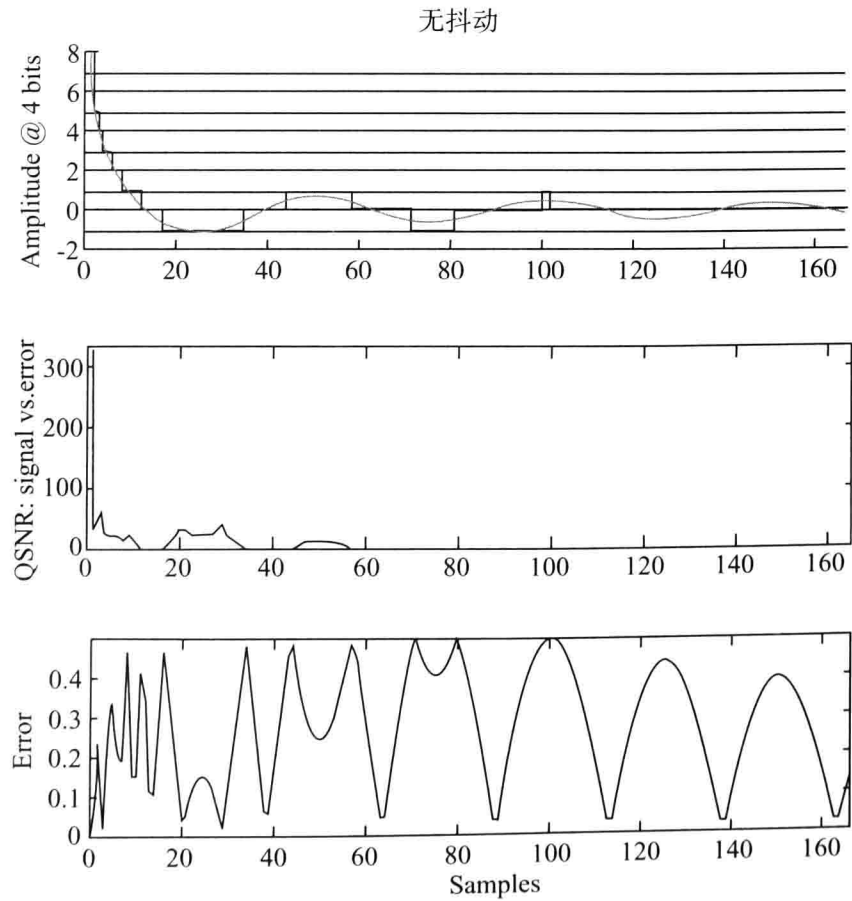


图 7.2 无抖动处理的四舍五入量化后的余弦波

会变成另一个完全不同的信号等问题。那么另一种解决部分此类问题的方法就是抖动。本质上,这是一种为低幅值信号提高 1 bit 范围,以间接提高比特分辨率的方法——它实际为更低的幅值提高了 1 bit 的限制域。

抖动是一种在采样和量化前,通过在原模拟信号中加入少量的误差信号来消除上述人工失真的简单而智能的方法。它本质上是使所产生的信号更吵闹,但所添加的噪声是如此之小,不管出于何种实际的目的,它都是微不足道甚至听不见的。经抖动处理后的信号被量化时,通过将模式化的类方波(quasi-square)转换为非模式化的波形而不会产生谐波失真特征。对于人耳,产生这种现象的一个主要原因是,我们往往会屏蔽掉噪声,并认为它是基底膜(人耳耳蜗的一个部分)上的背景噪声(尤其是当噪声音量较小时)的一般性质所导致的重复模式感觉,尽管在相对较高的本底噪声下这种模式化的噪声会显现出来。这样,从表面上看我们能够得到一个结论,即用抖动可以去除谐波失真,从而用低振幅的噪声使这些人工失真变得更不易察觉。

抖动的图解如图 7.3 和图 7.4 所示。图 7.3 的第 1 个图说明在极端情况下,不经过抖动处理,模拟余弦波很容易变成一个方波,这些方法为数字化后的信号带入了原模拟信号中所没有的谐波。另一方面,经抖动处理后的余弦信号如图 7.3 的第 2 个图所示,量化后,这种方波化重复模式减轻了很多。最后,图 7.4 给出了一个指数衰减的余弦波采用与不采用抖动预处理的量化结果。很明显,经抖动再量化的结果很好地消除了方波化模式。应用于上述图形的量化方法也称为脉冲宽度调制 PWM(pulse width modulation),指的是脉冲宽度的缩放与信号一致。

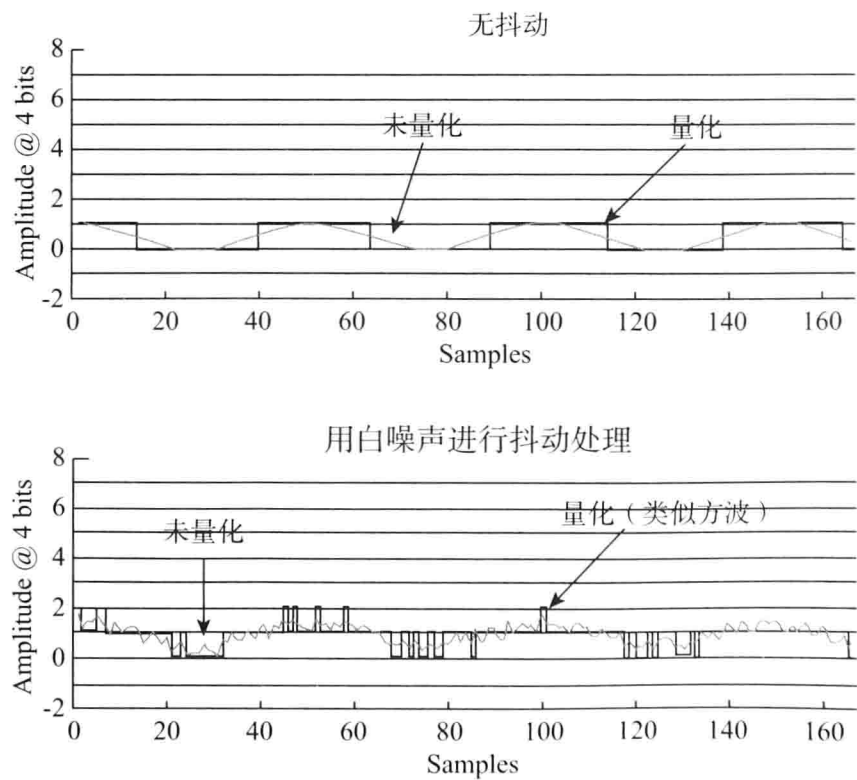


图 7.3 用抖动处理消除余弦波量化后谐波失真示意图

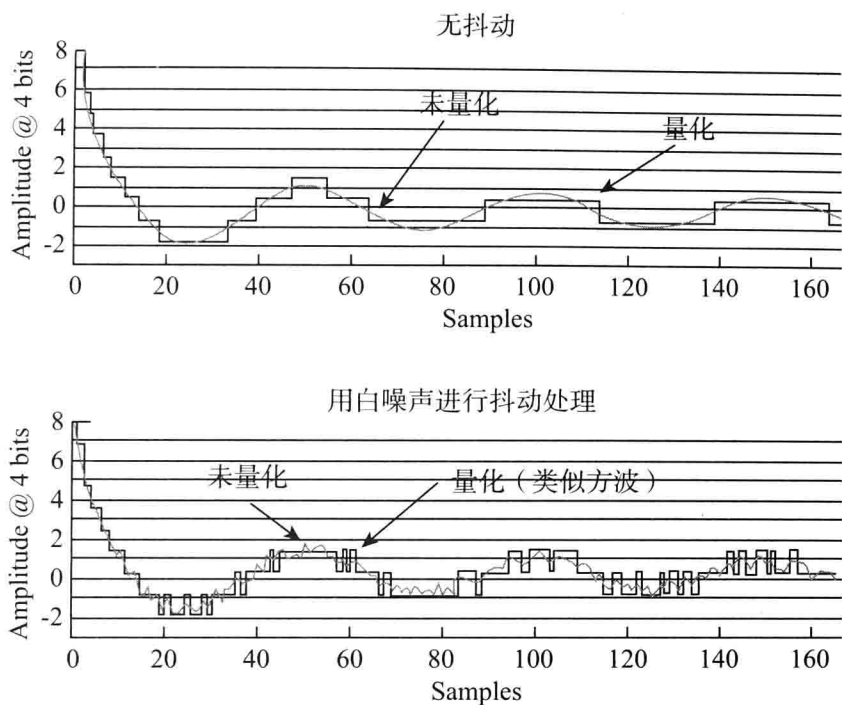


图 7.4 用抖动处理消除指数衰减余弦波量化后谐波失真示意图

8. 乐 例

有很多令人感兴趣的音乐实例得益于本章所论述的一些概念。一首典型的作品是来自于德国作曲家卡恩海因茨·施托克豪森创作的《联系》(Kontakte, 1958 ~1960)。在这首作品中,施托克豪森试图体现不同物体之间的声学特征,他将一系列不同的、相似的、变形的声音“联系”起来,并将频率从可数出周期数的低频脉冲慢慢提升至能逐步感觉出节奏、粗糙感(音色)和音高的高频脉冲,进而以此方法来探索音高、频率和粗糙感之间的界线。《联系》也是第一首采用多声道制作的音乐作品,它涉及到一种四声道立体声(quadraphonic)重放方案,即作曲家通过安装在可调旋转立柱上的扬声器来重放声音,然后用四只独立传声器将声音录制成一个四声道磁带。用这样的装置,施托克豪森获得了声源沿圆周运动的特征,它可以通过四声道立体声系统重放给听众。随着磁带录音机的出现(也是数字音频工作站出现的征兆),作曲家皮埃尔·斯奇弗(Pierre Schaeffer)在 1963 年创作了很多不可思议的、仅由各种开关门声和叹息声组成的音乐作品,名为《门与叹息声的变奏曲》(Variations for a Door and a Sigh)。在该作品中,作曲家通过操纵磁带制造时间延展和压缩(磁带机的快进和慢进播放),呈现了声音内在的音乐性——如爆炸节奏、音色和旋律性特征。斯奇弗给听众带来了与众不同的听觉体验。这首作品也是“具像音乐”(music concrète)中“寻音”(found sounds)手法运用的典型范例。约在 20 多年后,加拿大作曲家约翰·沃斯伍德(John Oswald)创造了“掠夺采样”(plunderphonics)一词,并撰写了题为《掠夺采样》(Plunderphonics)又名《盗版音乐是寻找创作素材的特权》(Audio Piracy as a Composition Prerogative, 沃斯伍德, 1986)一文,随后创作了大量改编作品,如《坏》(Bad, 以迈克尔·杰克逊的歌曲《坏》为素材),结果触及了版权和知识产权等敏感事宜,随后 CBS 公司和迈克尔·杰克逊对加拿大唱片工业协会进行了起诉。有趣的不是版权、复制权或者诉讼的复杂性,而是来自于原曲的素材被切成了只有细碎的一两秒钟甚至更短的片段。后来,作曲家剔除

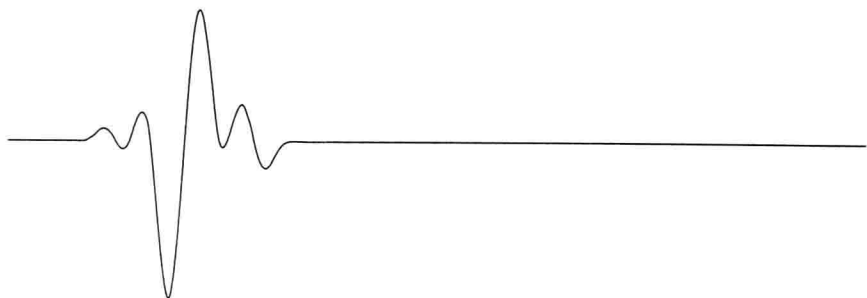
了这些短小的素材和片段,建立了节奏和节奏型,进行了大量重新创作后在本质上制作了一首全新的作品,但同时却保留了一些原曲独特的音质痕迹,以挑战听众对于音乐内容与语境、原版与再创作和新奇本身的体验极限。

这种采样和计算机音乐的另一个伟大范例是约翰·阿普尔顿(John Appleton)的《纽瓦克机场的摇滚》(*Newark Airport Rock*, 1969)。该作品被认为是作曲家在机场就电子音乐问题对乘客进行的访问采样。某种程度上,作曲家的任务变成如何从大量的采访录音中选取有用的声音进行组织、并置、重复,以形成一首具有音乐性的作品。这首作品也组织成了一部短小的文献,并对音乐结构本身进行了清晰地解读。计算机音乐家和计算机音乐的研究者常常游走于科学与艺术世界的边缘,当使用那些并不意味着必须使用的、新设计的合成算法或信号处理概念来进行创作时,这种写照会变得更加真实。这类作品的另外一个例子是笔者在 2000 年创作的作品《机器停止》(*The Machine Stops*)。这首作品刻意“滥用”本章所介绍的数字信号处理概念——如频率混叠、采样、失真和正弦波,它在音乐的语境中使用了如频率混叠的人工失真、采样、大于 20Hz 的采样盘(audio rate panning)和其他处理方法。乐曲开始处仅仅用一个纯音作为作品的关注动机——我们能用一个正弦波和一些“刻意被滥用的”基本数字信号处理技术来做什么?对于音乐创作,这是一件精彩的事情——不论结果是有趣的、令人满意的、吸引人的、烦闷的、又或是激烈冲突的,这些都无所谓,作曲的手法本身是没有错的。从工程的角度看,虽然这些手法对声音的处理具有破坏性,但作曲家却拥有打破工程常规、不计错误后果而进行创作的特权,他们可以随心所欲地使用这些技术,只要这些做法不伤害他人!

参考文献与扩展阅读

- [1] Dodge, C., Jerse, T. A. 1985. *Computer Music Synthesis: Synthesis, Composition and Performance*, New York, Schirmer Books.
- [2] Fletcher H., W. A. Munson 1933. “Loudness, its Definition, Measurement and Calculation”, *Journal of Acoustical Society of America*, 5, 82–108.
- [3] Oswald, J. 1986. “Plunderphonics, or Audio Piracy as a Compositional Prerogative”, *Musicworks* 34.
- [4] Zwicker E., Fast H. 1999. *Psycho-acoustics, Facts and Models*, Springer-Veriag Series in Information Sciences.

第 2 章 时域信号处理(一)



1. 引 言

接下来的两个章节将介绍时域信号处理的重要概念和技术。时域信号处理一词通常指在振幅随时间变化的二维空间中对信号进行分析、合成和调制等处理,如第 1 章所述,正弦波即是一个典型的二维时域信号。与时域相对的是将要在第 8 章讨论的频域。在基础知识准备阶段,频域信号处理理论也许相对时域来说较为困难,因为我们可以很自然地听到一个时域信号,因此在介绍频域信号处理理论之前,先花一些时间来介绍时域信号处理的知识是很有必要的,读者可以以此来熟悉数字信号处理的各种概念和方法。本章的内容包括振幅包络计算(amplitude envelope)、基频检测(pitch detection)、自相关分析(autocorrelation)、加和理论(add concept),以及一些经典的声音合成算法。按照惯例,本章仍会给出一些音乐作品分析实例。

2. 振幅包络与 ADSR 结构

所谓声音事件的振幅包络是指信号振幅随时间变化的形状或轮廓,也许到现在仍然是直接从声音波形获取某段时域总体特征的最为直接和重要的方法。有时候可以缩小波形,在损失细节的同时来观察振幅的整个轮廓。振幅包络是乐器尤其重要的音质特征,特别是通常所提及的 ADSR 过程,即 A 为激励阶段(Attack),D 为初始衰减阶段(Decay),S 为稳态持续阶段(Sustain),R 为释音残响阶段(Release)。正是这四个阶段将振幅包络分为四个基本区域,如图 2.1 所示。

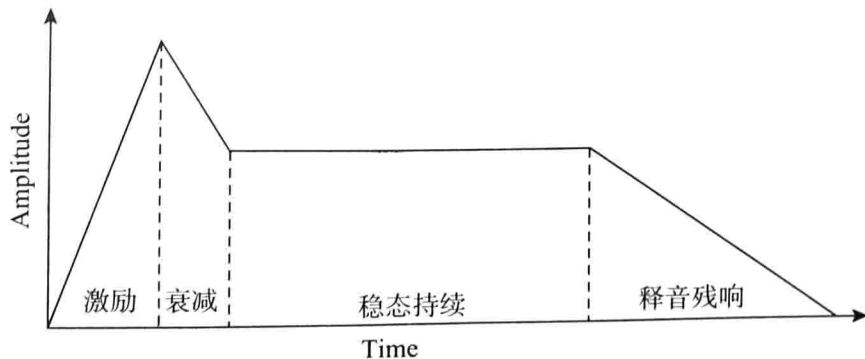


图 2.1 ADSR 包络

图 2.2 给出了斯坦威钢琴 C4 音的振幅包络,采样率为 44.1 kHz,量化深度为 16 bit。该图的上图为波形,中图为均方根 RMS(root mean square)振幅包络(下节详述),下图为 RMS 值的对数振幅包络曲线。由图可知,此乐音有着尖锐的激励、较长的稳态时间和释音时间。图 2.3 是小军鼓单击强音(fortissimo)的振幅包络,采用和上例相同的比特深度和采样频率。此例中,小军鼓声衰减得很快——300 毫秒处小鼓声已经衰减至 -40 dB,而同时钢琴声才衰减至 -18 dB,到约 4 秒附近才接近 -40 dB。振幅包络和 ADSR 结构如此重要的原因之一,是调整 ADSR 参数能够改变声音事件的听感——改变的程度取决于包络调制的类型与特征。有时甚至是振幅包络上相对很小的变化都能立刻改变声音的听感,极端情况下可能无法识别这种调制后的声音。请读者尝试在 MATLAB® 中播放一个钢琴和弦音或单音,然后在不改变参数的情况下试着播放反转后的声音,即由最后一个采样点开始往第一个采样点播放(代码 2.1)。此时肯定所有的参数都反转了,而不仅仅是振幅包络,而在所有参数都简单地进行时域反转处理后,声音事件的听感变得非常不同。

```
[x, fs] = wavread ('nameOfPianoSample.wav'); % read wave file
sound (x, fs)

disp ('Type any key to continue')
pause

xReverse = flipud (x); %flip the array up -> down
sound (xReverse, fs)
```

代码 2.1 声音的时域反转

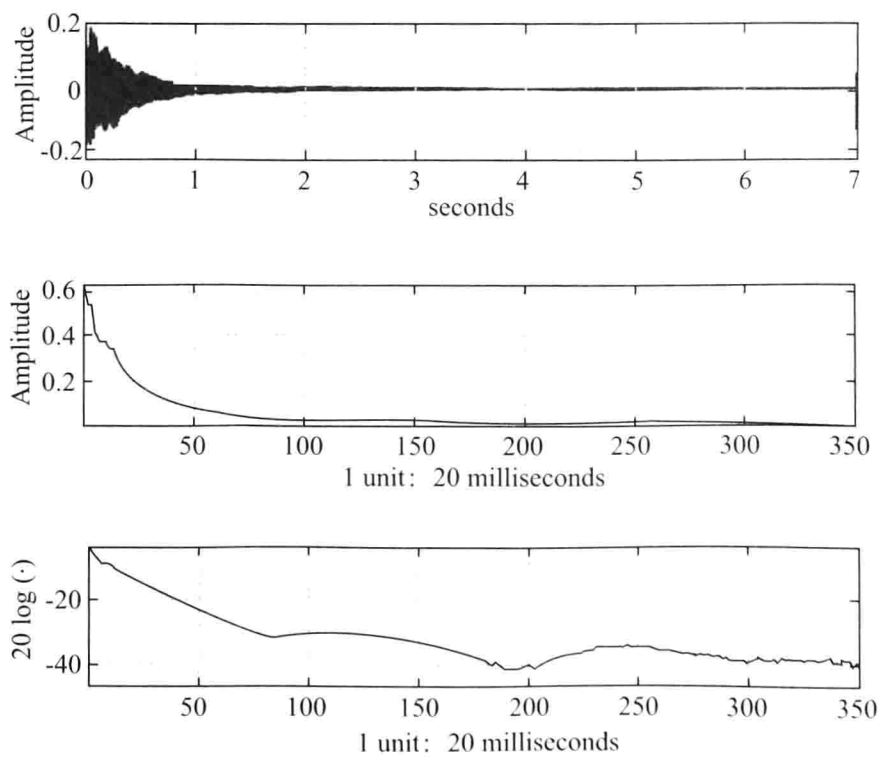


图 2.2 斯坦威钢琴 C4 音的振幅包络
(上图为波形,中图为线性包络,下图为 dB 对数包络)

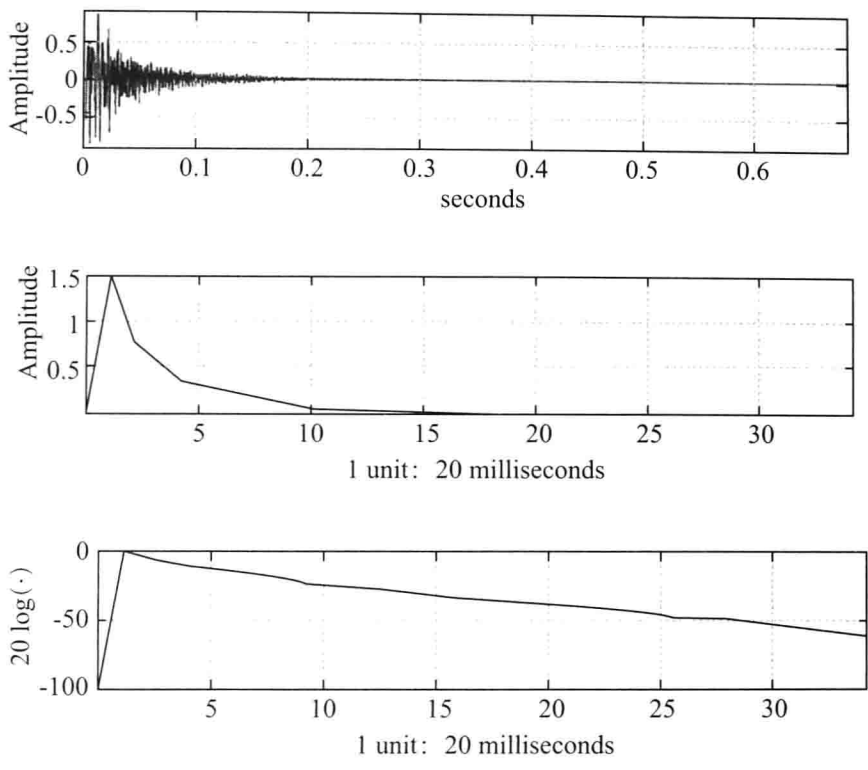


图 2.3 单击小军鼓的振幅包络
(上图为波形,中图为线性包络,下图为 dB 对数包络)

3. 波表合成

观察乐音的 ADSR 结构可以得到一些有趣的信息,如钢琴、颤音琴,或电贝斯都有着某些相似的 ADSR 结构,特别是在激励和稳态阶段。一般来说,波形的激励阶段即使不完全是随机的或“杂乱的”(noisy),也是复杂混乱的。激励阶段往往决定了乐音由开始衰减至音尾的总能量。而乐器乐音的稳态阶段通常听起来更加稳定,并显示出准周期(quasi-periodicity)的特征。图3.1给出了一个钢琴乐音中不同时间点的采样片段,从中可清楚地看到越接近音尾波形变得越平滑,也越表现出周期性。

一类用上述特征进行时域波形合成的方法称为波表合成(wavetable synthesis)。这一方法只存储真实乐器乐音波形的激励部分和一小段稳态部分。这样储存在只读存储器 ROM 中的波表只包含整个信号的部分信息——即完整的激励阶段信息和部分稳态信息。原因之一是出于经济的考虑,这样就可以节省合成器中的 ROM 空间以储存更多的声音,并使合成器更加强大且能够负担得起。读者也许会担心这些存储的有限激励和一小段稳态声音难以满足真正的作曲和演出需要,确实,也许重放一个短于或等于已存储波形的短促断音时,这一做法是可取的,但如果用户想延长声音的时长,并超过储存在 ROM 中的片段长度时,就会遇到已有波形无法满足既定播放要求的问题。而解决这一问题就是波表合成算法本质上所要做的事情。

在没有更妥当的术语时,可以用“稳态”(steady-state)来表述稳定的准周期阶段,如图 3.2 所示,这个概念也是指足够长的、能还原声音振动模式的阶段。波表合成要做的就是通过如图 3.2 所示的循环起始点(loop start)和循环结束点(loop end)之间的片段来合成声音。

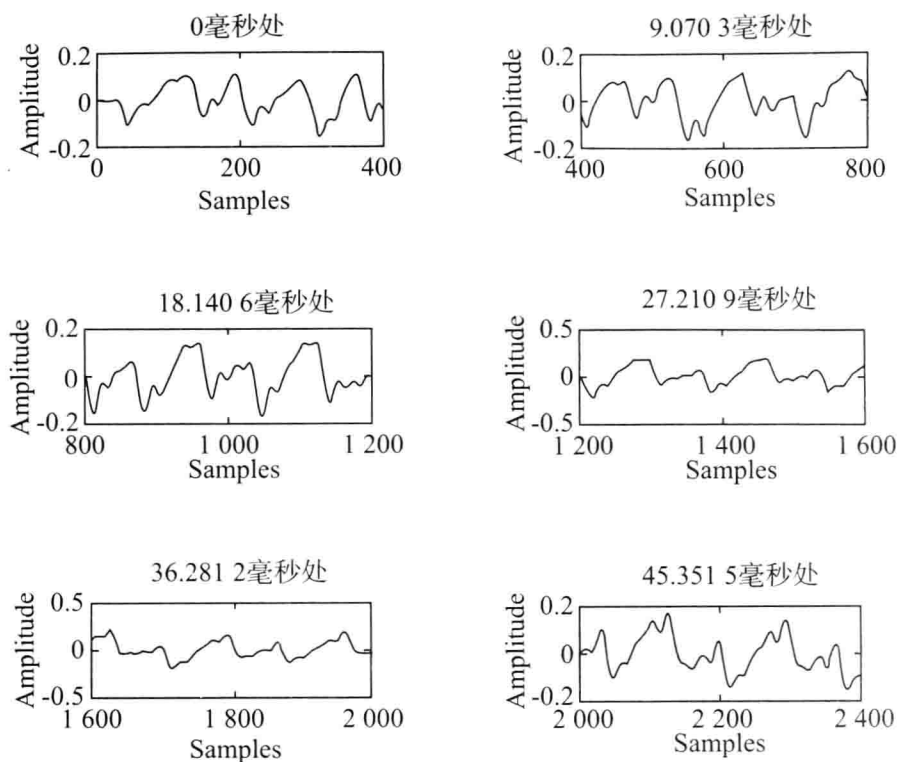


图 3.1 一个钢琴乐音中不同时段采样片段

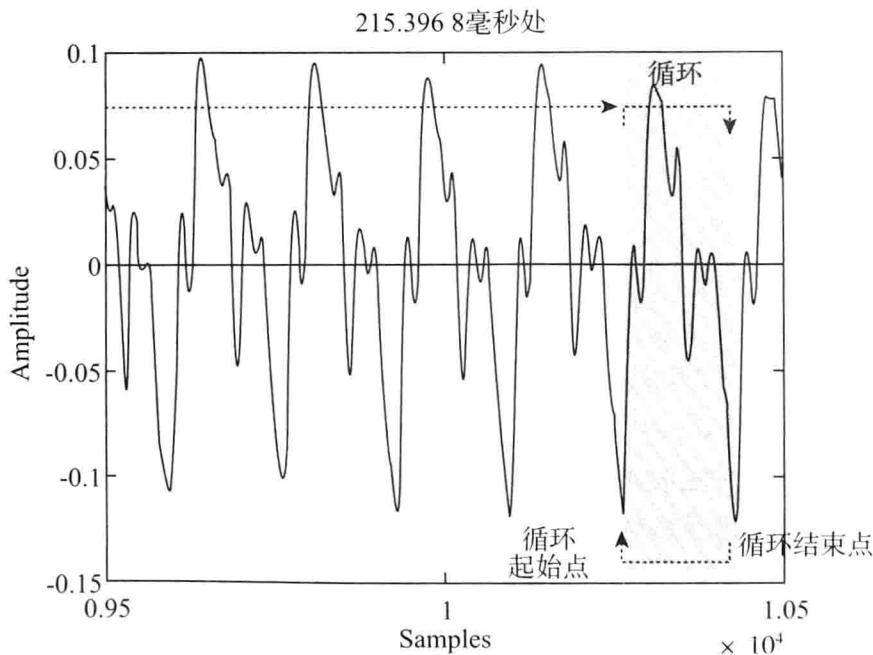


图 3.2 波表中的循环点

到达循环结束点时(即在激励和一小段稳态声音之后),合成器会保持循环在起始点和结束点间延续。通过这样的循环,人耳就会认为声音一直持续而没有终止。如果进一步在循环过程中通过包络慢慢降低振幅,那么即使用一个原声音的片段也可模仿原波形逐渐释音的残响特征。该钢琴乐音的总时长实际需要 108,102 个采样点,而在合成器的波表中大约只保存了该音 3,500 个采样点信息,仅为原信号的 3.2%,大大节省了储存空间。简而言之,波表合成器就是使用激励信息和部分稳态信息以循环片段的方式来合成声音的。但实际运用中,波表本身的信息

或许很难满足千变万化的音色。在波形合成器中,一个提高音色质量的方法是如何找到准确的、尽可能短的循环片段。另一个方法则是在打开或关闭滤波器来提亮或减暗声音之前,使用振幅包络来控制上述 ADSR 参数。

4. 时窗分析,RMS 与振幅包络

本节将从被称为时窗化(windowing)的重要概念开始,介绍一种与图 2.2 及图 2.3 相似的振幅包络计算方法。分析信号时,不可能对采样点进行逐个分析,特别是对于一个信息量巨大的信号,尽管实际上是一个整体,但不可能从头到尾把它作为一个单元来分析。基于逐点分析信号过于细致,将信号作为整体进行分析并且用一个、两个或几个分量来重构信号也是没有意义的。想象我们是如何听声音的——我们既不会按逐个采样点去听一张 CD(CD 的采样率是每秒 44,100 个采样点),也不会把整首歌曲从头至尾当做一个整体来听,而往往会把一个声音分成多个更小的片段。通常分析信号时,会将一系列采样点经过“时窗化”处理后再进行分析。例如对于一个有 10,000 个采样点的信号,使用 1,000 个采样点的“时窗宽度”(window size)。在这种情况下,先将第 1~1,000 个采样点时窗化,先分析第一组中的采样点,再分析从第 1,001 个采样点开始的下一组中的 1,000 个采样点,如此一直分析至信号的结尾处。这种一次提取一系列相邻的采样点作为一组进行信号分析的方法称为“时窗分析”(windowing)。取决于时窗宽度和信号长度的多个时窗分析结果和这些时窗都是相互独立的,组合起来可形成反映整个信号的总体结果。

4.1 时窗分析的进一步讨论

最常用的时窗是“矩形窗”,其陡峭边缘的时域波形如图 4.1 所示。矩形窗在本身的矩形范围具有单位增益,其他区域的值为 0,如式 4.1,其中 N 与时窗宽度有关为整数, n 为采样点序号, $n \cdot T$ 是单位为秒(s)的时间位置。

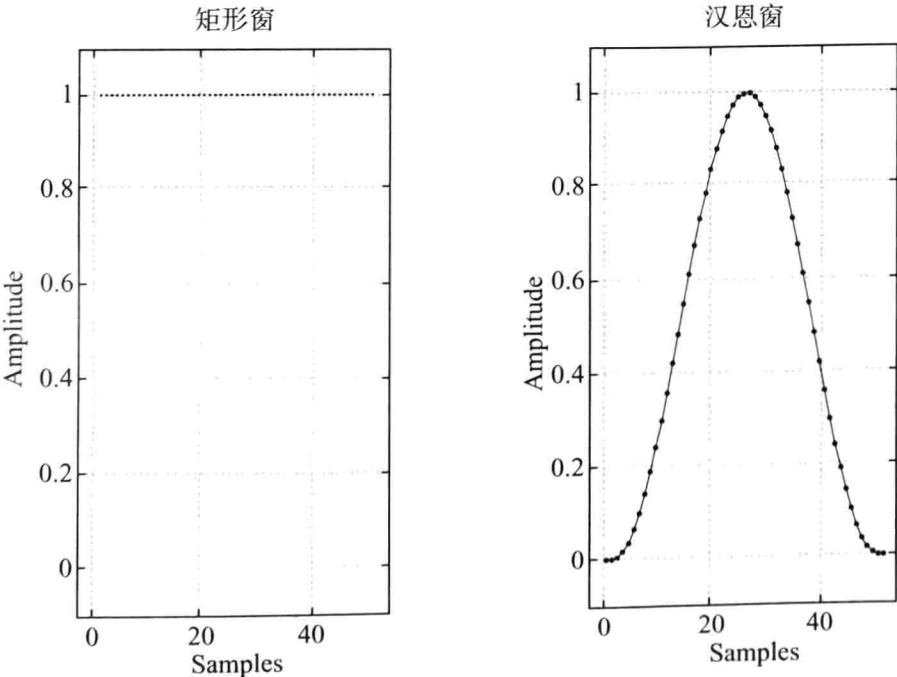


图 4.1 50 个采样点宽的矩形窗和汉恩窗

$$w[n] = \begin{cases} 1, & 0 \leq n \leq N - 1 \\ 0, & \text{其他} \end{cases} \tag{4.1}$$

如前所述,时窗分析是提取信号的某个部分进行分析,准确地说这一部分就是所得到的结果。而当我们说对一个信号进行窗口化时,实际在技术处理上是用该时窗函数乘上一部分长度与时窗宽度相等的信号,如下式所示:

$$x_{win}[n] = w[n - L] \cdot x[n] \tag{4.2}$$

该式中, L 是用整数表示的时间偏移量(也称为时延),它间接地确定了波形 $x[n]$ 的起始采样点序号——而在时窗区域以外的信号则乘以 0。

如果上式还存在一些混淆,代入一些数值也许会有助于让事实变得清晰。代入 L 的值,则当 $N=50, L=0$ 时式 4.1 可改为:

$$w[n - L] \big|_{L=0} = w[n] = \begin{cases} 1, & 0 \leq n \leq 49 \\ 0, & \text{其他} \end{cases} \tag{4.3}$$

保持 N 的值为 50 不变,令 $L=1$ 有:

$$w[n - L] \big|_{L=1} = w[n - 1] = \begin{cases} 1, & 0 \leq n - 1 \leq 49 \\ 0, & \text{其他} \end{cases} \tag{4.4}$$

不等式 ($0 \leq n - 1 \leq 49$) 的左右两边各加 1,式 4.4 可改写为式 4.5,所得到的中间这部分区域就是所关注的时窗宽度。

$$w[n - 1] = \begin{cases} 1, & 1 \leq n \leq 50 \\ 0, & \text{其他} \end{cases} \tag{4.5}$$

因此,可平移时窗的一般形式定义为:

$$w[n - L] = \begin{cases} 1, & L \leq n \leq N - 1 + L \\ 0, & \text{其他} \end{cases} \tag{4.6}$$

其中 L (长度为 L 个采样点的时延)只是在时间轴上将时窗平移 L 个采样点,以选取一段时窗化后长度为 N 的信号。简而言之,对于矩形窗,“平移”表示在需要分析信号范围内提取、剪下或切取一个波形片段,而保持信号幅值不变,如图 4.2。

以下代码实例给出了一段用矩形窗对从硬盘读取的钢琴声信号进行简单分析的例子。

```
[x, fs] = wavread('nameOfPianoSample.wav'); % read wave file
xWindowed = x(1:500);
sound(xWindowed, fs);
```

代码 4.1 矩形窗分析实例

也许会在矩形窗提取信号的起始点和终止点听到“叭叭”声,虽然这种声音会令人不悦,但从本质上看,这并不是时窗分析误差造成的,而是归因于该时窗陡峭的边缘。这种现象使我们不得不介绍另一种时窗——汉恩窗(Hann window),有时也称为汉宁窗(Hanning window)。正如图 4.3 的中图所示,汉恩窗有平滑的边缘,峰值位于波形中央,幅值向两侧逐步衰减。这种类型的时窗在

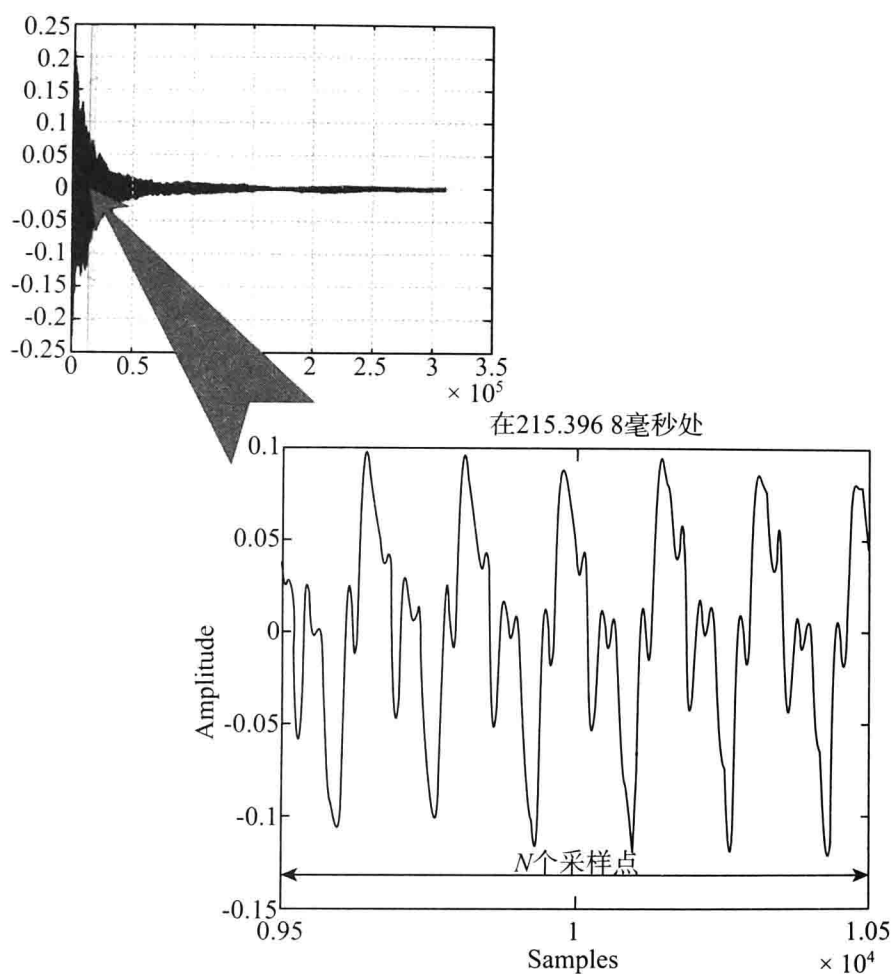


图 4.2 钢琴声采样的时窗分析

对信号进行分析时不会听到由矩形窗所引起的“喇叭”声,但会在时窗化后的片段内改变原信号的波形特征,图 4.3 的下图显示了原信号与汉恩窗截取后信号之间的差异。

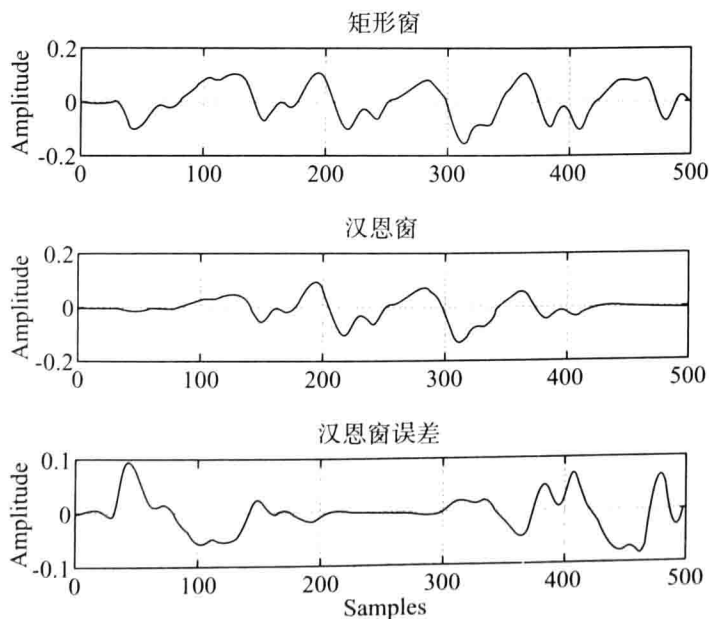


图 4.3 矩形窗(上图),汉恩窗(中图),汉恩窗误差(下图)

除了波形的幅值发生改变以外,在没有工具分析信号的频域特征之前,现在很难说哪一种时窗的形状更适合用来分析其所相乘的波形。不过,一般规律是边缘陡峭的波形比边缘更平缓或更平滑的波形要复杂。直观上看,众所周知,噪声信号在所有时域位置上都有分布,采样点与采样点之间的幅值变化往往都很剧烈,而正弦波则相反,采样点与采样点之间的幅值变化很平滑,没有陡峭边缘。因此可以说,相较于汉恩窗,矩形窗会带来一些人为失真,这取决于时窗的边缘特征和形状。第 8 章还会再介绍各种时窗类型的重要特征,并讨论人为失真的种类,以决定在何场合下使用何种类型的时窗。

4.2 RMS 与振幅包络

简单地说,振幅包络就是对一个波形的幅值进行宏观检测。一种常见的计算振幅的方法是 RMS 算法。RMS 是“均方根”(root mean square)一词的简写,其定义如式 4.7。顾名思义,算法中的 L 为时窗宽度, n 为采样点序号。

$$\text{RMS} = \sqrt{\frac{1}{L} \sum_{n=0}^{L-1} x^2[n]}$$

(4.7)

RMS 值位于信号峰值和算术平均值之间的某个位置。甚至是在信号强度较高而没有太好的瞬态响应(transient response,与强度变化有关的响应)的情况下,该均方根值也相对稳定。但有一个与人耳听觉感知方式不同的极端情况,即每一个采样点的值都在变化,从而导致包络细节和瞬态信息过于丰富。而 RMS 值更接近于人耳听觉系统对音频信号强度变化的敏感程度。图 4.4 给出了正弦波及其 RMS 值、峰值和峰对峰值(peak-to-peak)的动态范围。

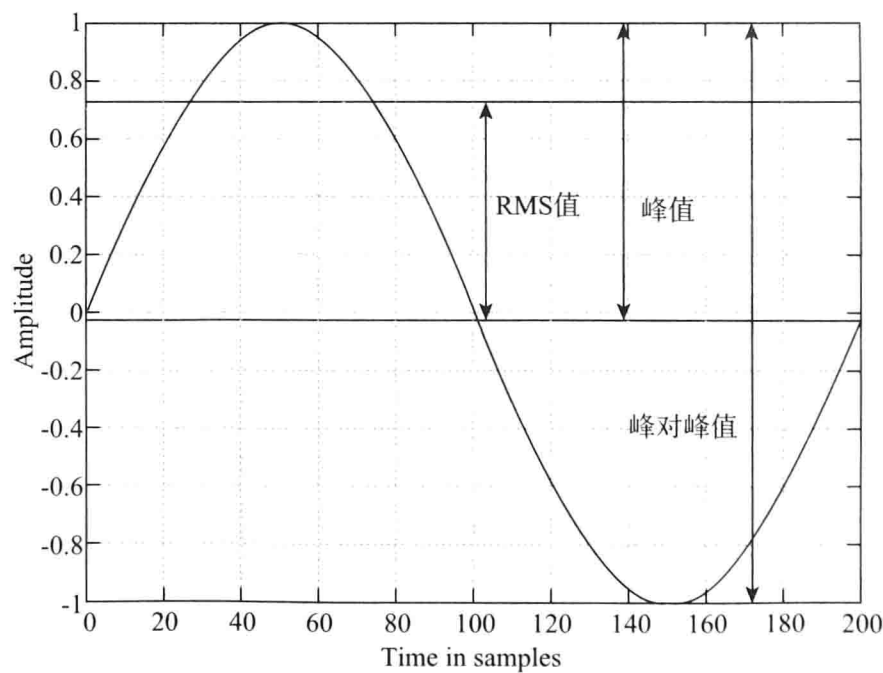


图 4.4 正弦波的 RMS 值,峰值与峰对峰值

用 RMS 来计算振幅包络时,需要对信号进行时窗分析,如图 4.5 和图 4.6 所示。由图 4.6 可知,时窗化后的结果使 RMS 包络长度比原信号要短。而这并不难理解,因为每一次时窗分析只反映时窗内所选取的信号片段的特征。时窗宽度越小则瞬态响应越好,时窗宽度越大则可观察更长时域内的信号,但瞬态响应越慢。如墙上所挂的纸质日历,一年有 365 天,而这种日历一般是一页一个月,每页约包含 30 天,这 30 天就可以作为一个时窗宽度来考察各月的特定特征——如一月最冷,八月最热(当然取决于住在地球的哪个位置)。不过这样会损失每一天的瞬态特征,或者说逐日和逐星期的变化细节。此时,瞬态响应成为时窗宽度的函数。时窗越长得到瞬态细节越少,时窗越短获得的瞬态特征就越多。那么问题就变成了合适的时窗宽度是多少。这当然是一个难于得到简单答案的问题。对于特定的分析目标,它取决于分析的具体情况和需要多高的分析精度。例如以一分钟作为时窗宽度,一页日历纸仅容纳一分钟的话,无疑过于细致了,虽然并没有什么情形会这样做,因为这样的话一本年历将需要大量的纸张!

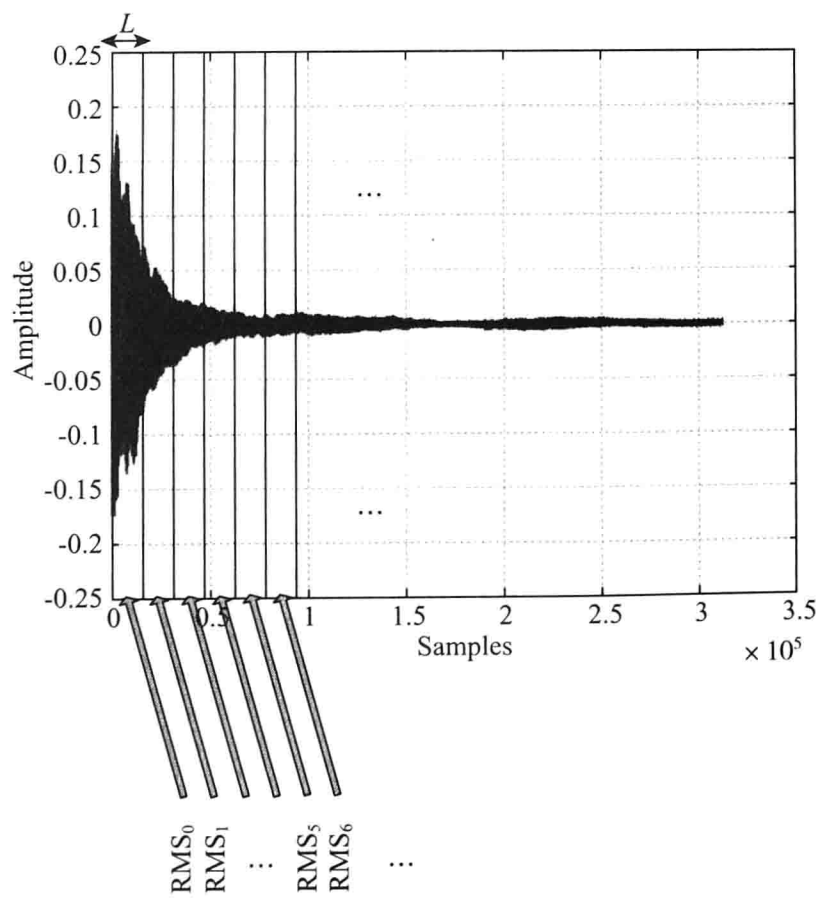


图 4.5 波形的时窗分析和 RMS 帧计算

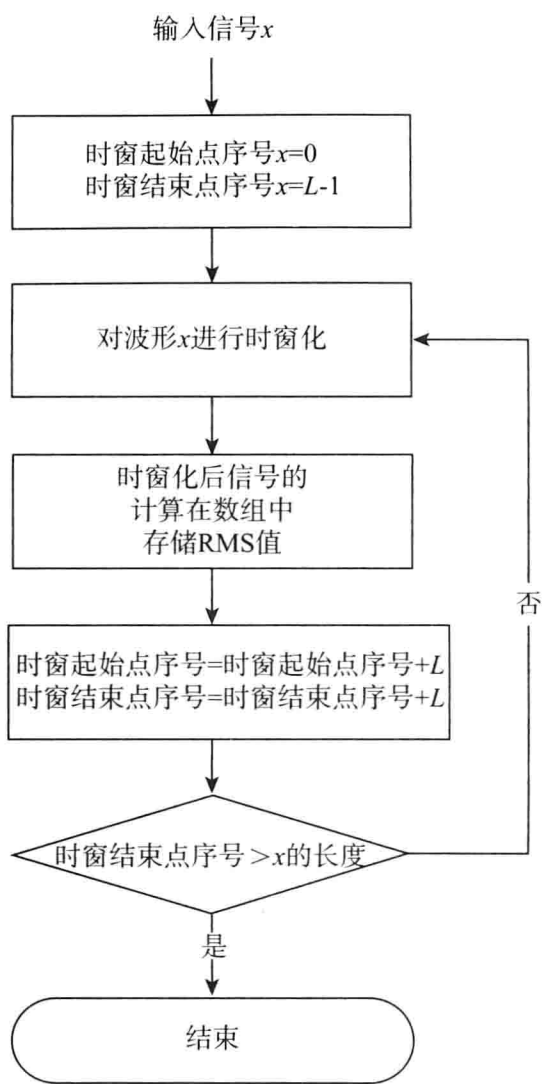


图 4.6 RMS 包络的计算过程

5. 时域内的基频提取方法

本节将讨论两种基本的音高算法。因为音高是对声音事件的周期或准周期的听觉感知,所以提取音高通常是指寻找声音的基频。本节将要介绍的两种常见的时域内检测音高的算法是过零率(zero-crossing-rate)和自相关函数(autocorrelation)。这两种算法相对简单易用,同时也非常有效。

5.1 过零率

过零率是一种直观的计算基频的方法。顾名思义,该算法沿波形计算两相邻过零点(zero-crossings,零点指 $y=0$ 轴上幅值为 0 的位置)之间的时间(用采样点的个数表示),如图 5.1 所示,其中黑点为幅值等于 0 或恰好位于零点之上或之下的采样点。

对于图 5.1 中上图所示的正弦波,过零点的位置和数量的计算如你想象的一样容易。基频的计算同样也很直观——赫兹定义为每秒内的周期数,则正弦波的基频计算如式 5.1 所示。即此例每秒 44 100 个采样点中有 522 个过零点,而对于一个理想的正弦波每周期有两个过零点,

则该正弦波音高所对应的基频为过零点总数除以每周期的过零点数,如式 5.1。

$$f = \frac{522\text{zcrs/sec}}{2\text{zcrs/cycle}} = 261\text{Hz} \tag{5.1}$$

也就是说,基频可用由过零点位置表征的每周期的采样点数来表示,则该基频可用如下式 5.2 计算。

$$f = \frac{44,100\text{samples/sec}}{170\text{samples}} = 259.41\text{Hz} \tag{5.2}$$

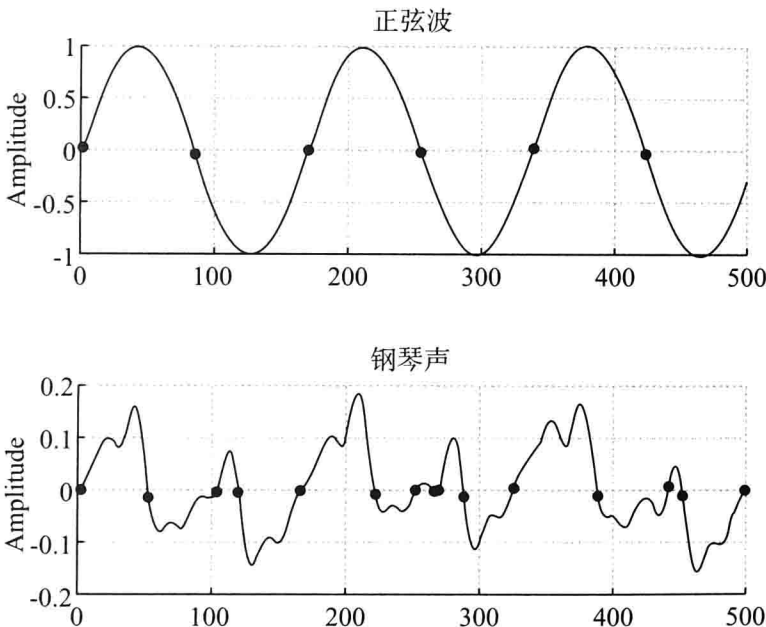


图 5.1 正弦波(上图)和钢琴声(下图)
[音高为 C4(261Hz), 采样率 $f_s = 44.1\text{ kHz}$]

到此为止,该算法似乎完美得让人有所怀疑,其实计算音高的过零率算法也有其自身的问题。图 5.1 下图中的钢琴声,波形的复杂程度高于正弦波。该波形的复杂度导致每周期的过零点数不同,并使该算法的结果产生误差。如该声音的第 1 个完整周期有 4 个过零点,第 2 个周期有 6 个,第 3 个周期有 4 个。尽管音高听起来是一致的,但波形每周期的过零点数有所变化。一般来说,波形越复杂,此类问题就越严重——而音乐信号往往都是复杂信号。解决这一问题的办法有很多,包括计算有限带宽内的过零点数以及一段时间内的平均过零点数。为了限制带宽,可限制较高和较低基频的上限和下限:

$$27 \leq \text{频率 } f \leq 4\,200 \tag{5.3}$$

代入每周期的采样点数,式 5.3 可写成式 5.4。

$$1633 > \text{采样点数} > 10 \tag{5.4}$$

上述频率对应钢琴上的最高音 C8 和最低音 A0。有了这种滤波器,间隔很短的成对过零点就可以忽略。如两个横跨 6 个采样点的过零点(对应于 7350Hz)便超出了上述范围,不能算是有效的过零点。

图 5.2 给出了和上例一样的正弦波(上图),以及两段不同时间段的钢琴声样本——钢琴声的起始阶段(中图)和稳态及释音阶段(下图)。由图可知,过零率算法 ZCR 更适合于分析该信号的尾部,这不奇怪,因为钢琴声呈准周期的稳态阶段更近似于一个正弦波。如图 5.2 的下图所示,平均过零率 ZCR 为 82 个采样点,非常接近同频率的正弦波(图 5.2 的上图)。也就是说,乐音的激励阶段具有最复杂的结构,与纯音波形相去甚远而与噪音更为接近,进入稳态阶段和释音阶段后,其波形就越接近正弦波。同时,高频能量降低,只有低频谐波分量持续到最后,而这些低频分量往往包含了基频。ADSR 的频域能量变化将进一步在第 8 章讨论。而现在,设乐器乐音的音高感知是由一系列既定谐波 f_k 所构成的,每个谐波在理想情况下都是基频 f_0 的整数倍,即 $f_k=k \cdot f_0, k$ 为整数。

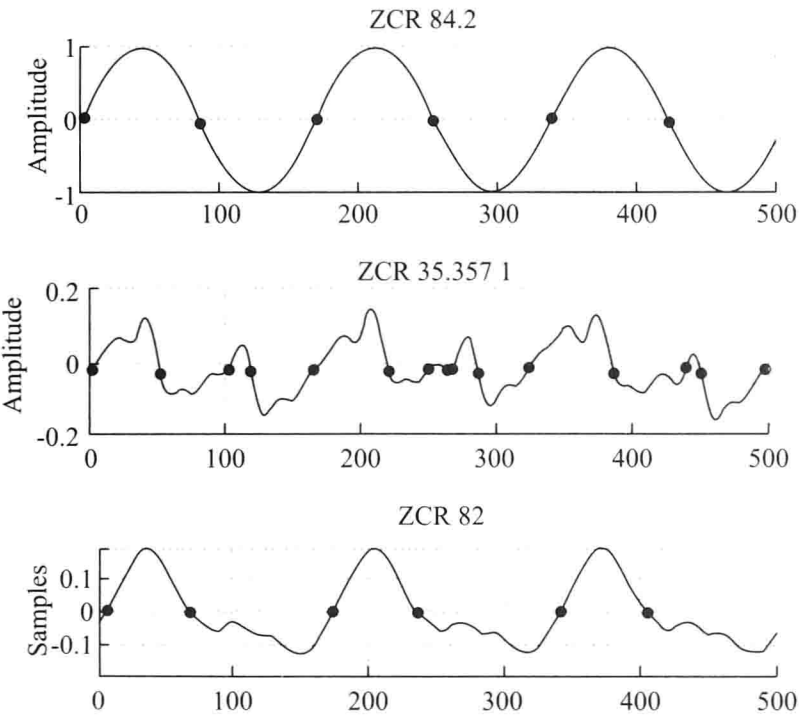


图 5.2 正弦波的过零率和过零点(上图),钢琴声的过零率和过零点(中图和下图)
[音高为 C4(261Hz),采样率 $f_s=44.1\text{ kHz}$]

另一个用 ZCR 算法来检测音高的方法,是先按前面提到过的方法计算一个类似于 RMS 的包络,只不过这次不是计算 RMS 值而是算术平均值。也就是说,可用一种改进后的 RMS 算法来替代式 4.7,如图 5.3,该算法的核心是算术平均值的计算。

图 5.3 中的框图给出了一个简单的计算过程:首先通过算术平均值来滤除波形细节以平滑信号,然后为了提高 ZCR 算法的鲁棒性(robust),降低时间分辨率,本质上相当于“时域模糊处理”(time blurring)。请注意该算法除了计算时窗截取的信号片段(帧)的算术平均值和最后使用插值器外,与 RMS 算法基本相同。这里,插值器只是用来增加采样点而满足原信号的原始长度。需要注意的是,对信号进行时窗分析会损失采样点的数量。例如用一个宽度为 10 个采样点的时窗来分析长度为 100 个采样点的信号,则尽管分析结果仅有 10 个信号帧,但总采样数却已经下降了十倍。

图 5.4 给出了用宽度为 50 个采样点(1.13 毫秒)的时窗来计算算术平均值的例子。这种信号平滑处理技术称为滤波,将在第 5 章正式讨论。由于该算法沿波形移动并计算均值(算术平均值),因而被称为移动平滑滤波器(moving average filter)。对比图 5.4 的上下两图可知,使用

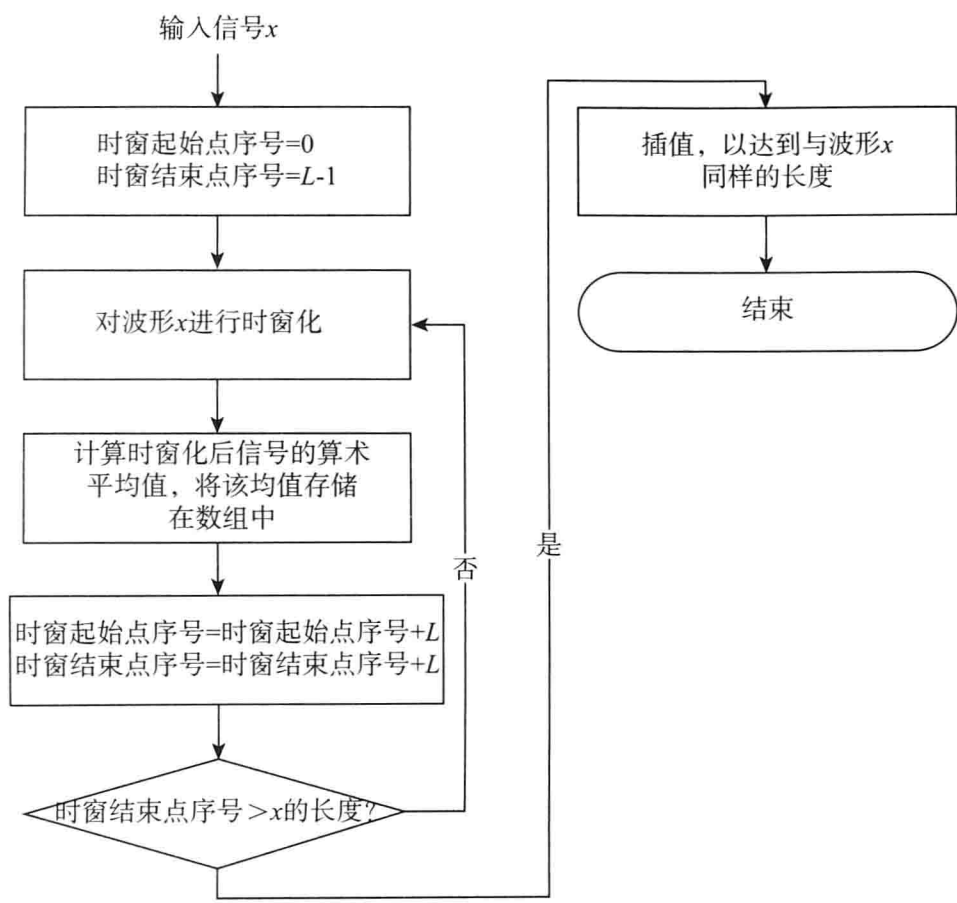


图 5.3 逐帧均值计算及其插值

这种改进的 ZCR 算法后, 音高计算的准确性有了极大的提高。

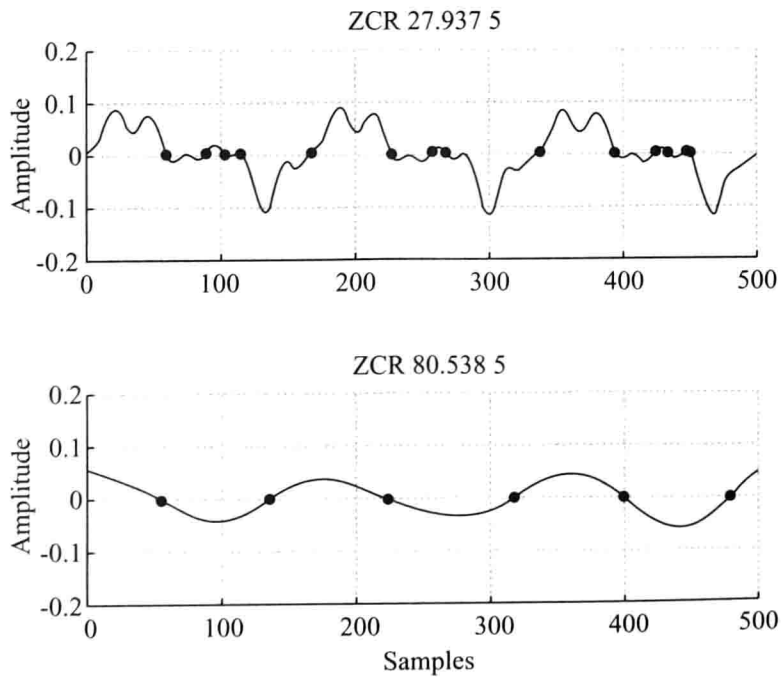


图 5.4 原信号的过零点位置(上图)和经过均值平滑和插值处理后的波形(下图)

5.2 自相关函数

自相关函数是另一种常用的时域检测音高的方法。该算法基于自相似性检测 (similarity measurement) 的概念,其乘积求和形式如式 5.5 所示。

$$acf_{xx}[\tau] = x[\tau] * x[-\tau] = \sum_{n=0}^{N-1} x[n] \cdot x[n + \tau] \tag{5.5}$$

式 5.5 中, τ 为延迟, $acf_{xx}[\tau]$ 是自相关函数的值, N 为帧长度 (即用矩形窗所截取的片段的采样点数), n 为采样点序号, 且当 $\tau=0$ 时, $acf_{xx}[\tau]$ 称为信号的功率 (即在特定时窗下信号样点的平方和)。由式 5.5 可知, 与 RMS 的计算方式类似, 自相关函数也是沿时窗所截取的信号片段平移, 每个时窗所对应的帧乘以其其他帧后累加求和。即保持其中一帧不变, 其他帧 ($x[n+\tau]$) 以时延 τ 对输入信号 ($x[n]$) 进行平移, 如此周而复始。

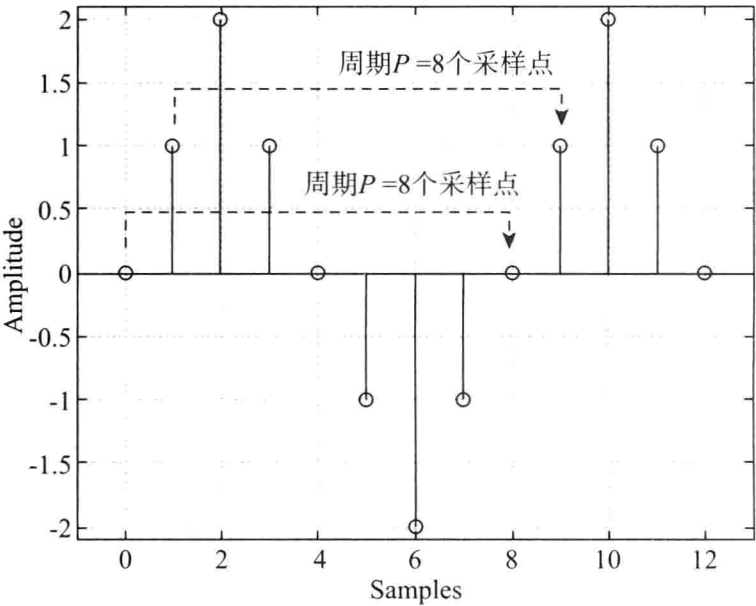


图 5.5 周期 $P=8$ 的正弦波

自相关分析意味着如果信号是周期性的, 则在一个完整周期后它自身会重复。因此, 以相似性而言, 时移位于周期或基频的整数倍时, 其对应的信号总是具有相似性的。以图 5.5 所示的一个低采样率的正弦波为例, 一个理想的正弦波形会在一个周期后重复, 此例周期 $P=8$ (即每隔 8 个采样点会重复)。那么在计算自相关函数时, 延迟 τ 由初始值 $\tau=0$ 开始沿正弦波平移并增大。下面让我们通过求解一些延迟 τ 的值来更好地了解自相关函数背后的意义。

$$\begin{aligned} ACF[0] &= \sum_{n=1}^2 x[n] \cdot x[n + 0] = \sum_{n=1}^N x[n]^2 = 1(1) + 2(2) = 5 \\ ACF[1] &= \sum_{n=1}^2 x[n] \cdot x[n + 1] = 1(2) + 2(1) = 4 \\ ACF[2] &= \sum_{n=1}^2 x[n] \cdot x[n + 2] = 1(1) + 2(0) = 1 \\ ACF[3] &= \sum_{n=1}^2 x[n] \cdot x[n + 3] = 1(0) + 2(-1) = -2 \end{aligned}$$

$$\begin{aligned}
 ACF[4] &= \sum_{n=1}^2 x[n] \cdot x[n+4] = 1(-1) + 2(-2) = -5 \\
 ACF[5] &= \sum_{n=1}^2 x[n] \cdot x[n+5] = 1(-2) + 2(-1) = -4 \\
 ACF[6] &= \sum_{n=1}^2 x[n] \cdot x[n+6] = 1(-1) + 2(0) = -1 \\
 ACF[7] &= \sum_{n=1}^2 x[n] \cdot x[n+7] = 1(0) + 2(1) = 2 \\
 ACF[8] &= \sum_{n=1}^2 x[n] \cdot x[n+8] = 1(1) + 2(2) = 5 = ACF[0]
 \end{aligned} \tag{5.6}$$

图 5.6 进一步给出了上例自相关函数计算过程的图解,其中帧长 $N=2$,延迟 τ 的值由 0 变化至 8,图 5.7 则是该自相关函数的计算结果。由图可知, $ACF[0] = ACF[8]$,如果继续计算 $\tau > 8$ 的自相关函数值会发现 $ACF[1] = ACF[9]$,则一般有:

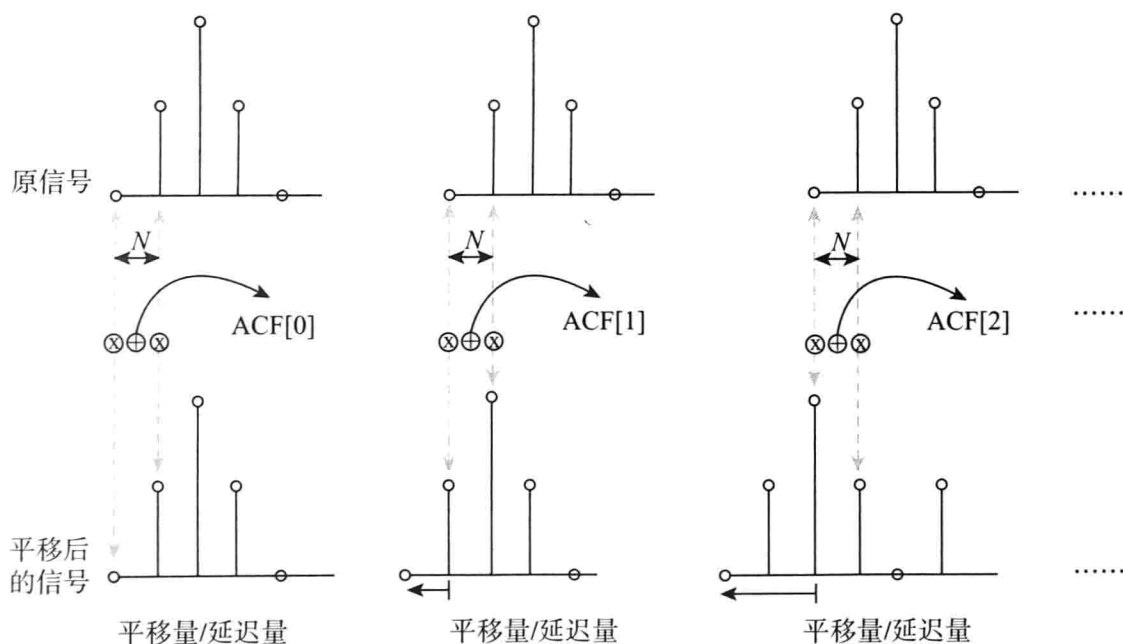


图 5.6 $N=2$, 时延 $\tau=0 \sim 8$ 的自相关函数计算示意图

$$ACF[\tau] = ACF[\tau + P] \tag{5.7}$$

也就是说平移 P 个采样点后,自相关函数达到最大值,这是表征信号周期的重要信息。因此,频率和音高(基频)可用式 5.8 计算,其中 f_s 为采样率,单位是每秒的采样点数, P 为一个周期内的采样点数。

$$f_{Hz} = \frac{f_s}{P} \tag{5.8}$$

图 5.8 是没有经过滤波的钢琴声信号的自相关函数计算结果,之前在介绍 ZCR 算法时曾使用过。该例可清晰而简便地通过峰值检测算法(peak detection algorithm)侦测基频($P=168$ 个采样点)。如在这种情形下可设计一个能量阈限为 40 的峰值检测器(peak detector),通过计算斜率极值的变化而得到峰值。

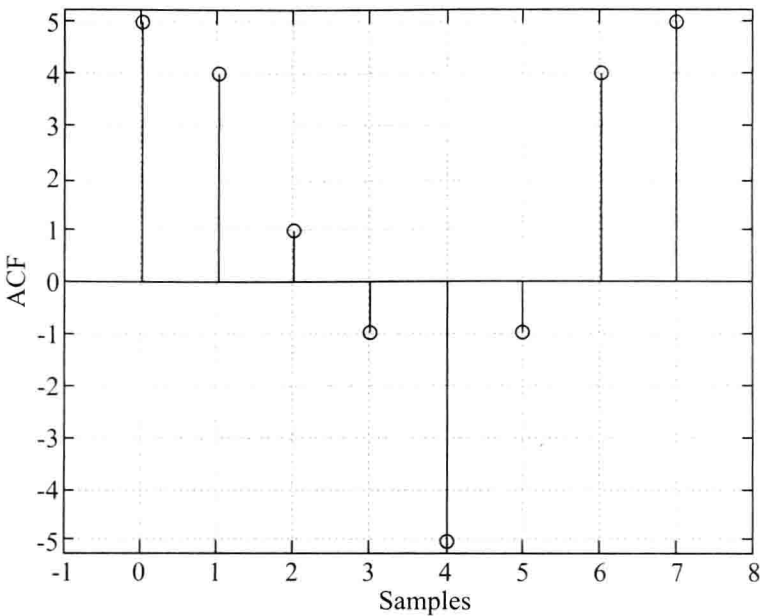


图 5.7 $N = 2$, 时延 $\tau = 0 \sim 8$ 的自相关函数图像

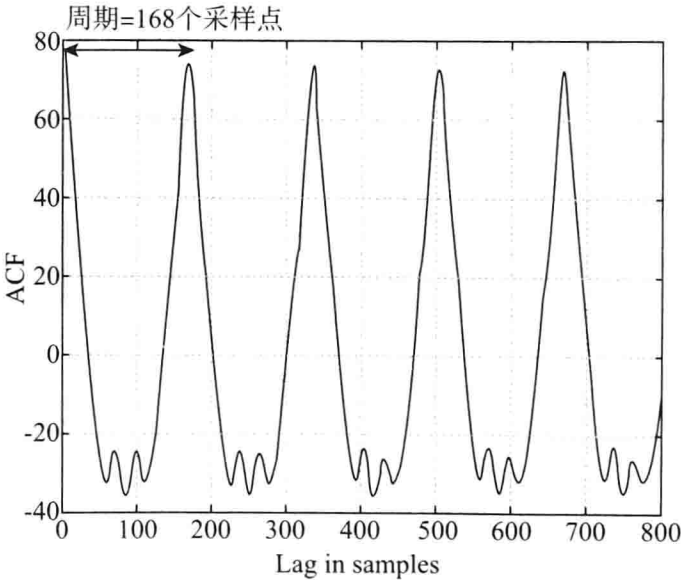


图 5.8 钢琴声的自相关函数

基于自相关函数的基频检测计算方法总结如图 5.9, 采样点间斜率可直接按照式 5.9 计算。

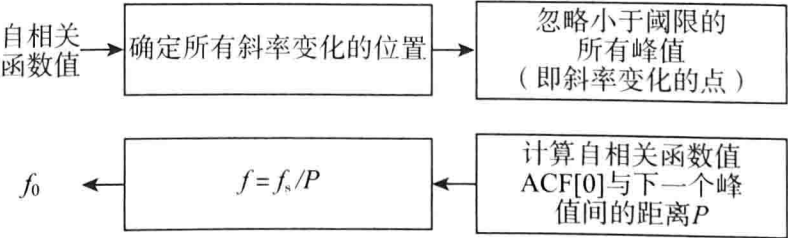


图 5.9 基于自相关函数的音高检测算法

$$\text{斜率} = \frac{x[n] - x[n - 1]}{2} \tag{5.9}$$

一般来说,帧长 N 越大,自相关函数的峰越明显,可供分析的数据就越多。而计算长时信号的自相关函数 ACF 时,帧越长瞬态响应却越弱。如果音高变化很快,使用长度大的帧就会因缺失瞬态特征而使音高检测的结果产生误差。因此,需要在瞬态响应与基频计算精度之间保持平衡。当然如果预先知道需要分析的是哪种乐器,也可以提高该基频检测器的准确性。如用自相关函数算法来设计一个吉他调音器,由于频率 f 与波长 λ 成反比,就不需要考虑贝斯吉他的频率,并让最大帧长尽可能短,以提高整体的瞬态响应。

另一个在自相关函数音高检测算法中提高瞬态响应的方法是动态地改变帧长。这一方法的基本思想是以最低基频来决定极大帧长。该极大帧只是用来检测陡峭峰值,如果计算所得到的周期内采样点数偏小则会导致所检测的音高偏高,即此时不需要再使用该极大帧长,而使用一个更短的帧长就足够了。其优点是无论何时,该算法都会选择一个较短的帧长从而提高瞬态响应。该算法的提纲如图 5.10。

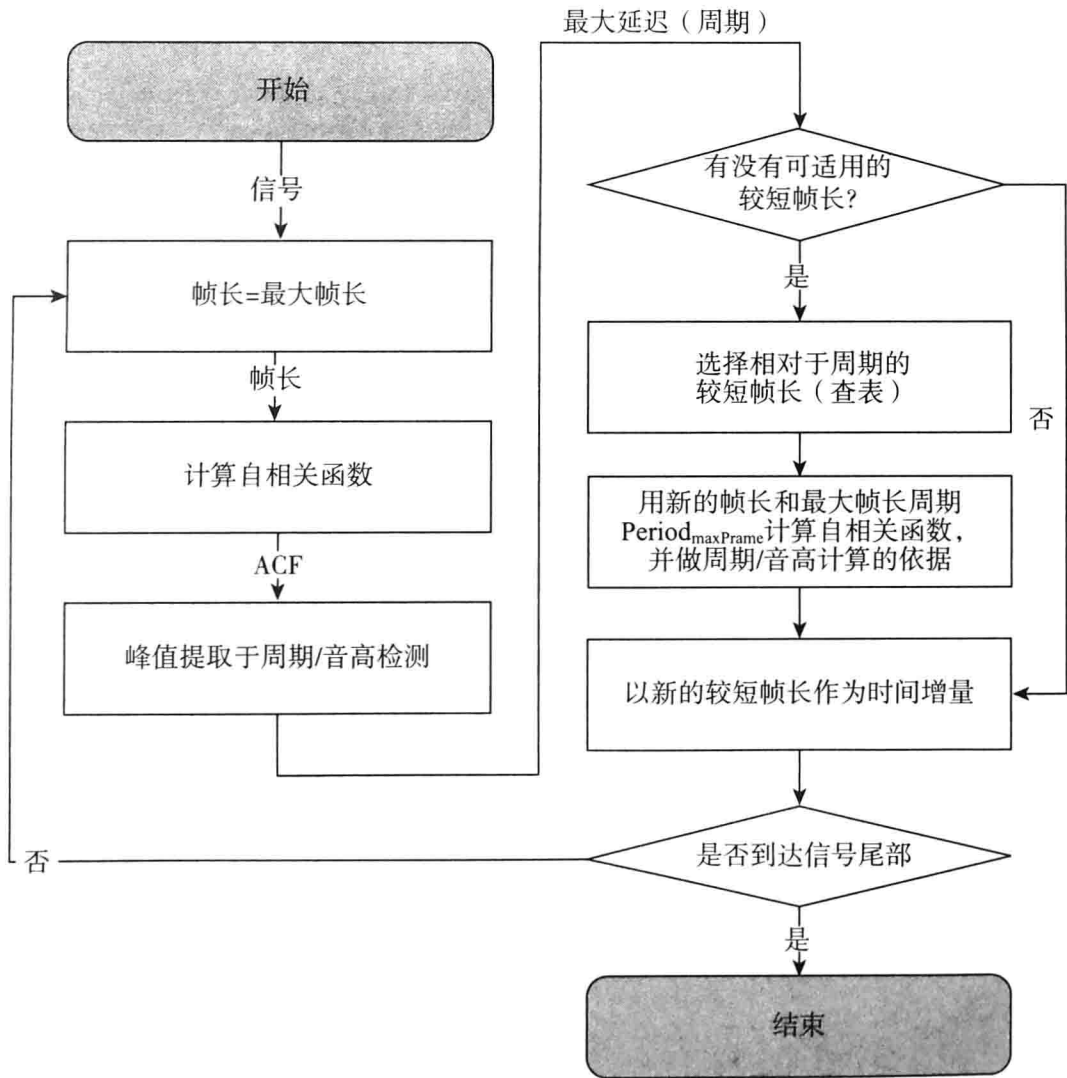


图 5.10 调整延迟来提高瞬态特性的算法流程(帕克,2000)

尽管在大多数情况下,自相关函数法由于良好的鲁棒性而被广泛用于移动电话、语音和音乐信号的分析中,但有很多情形它也无法准确地计算基频。这种情况往往出现在信号自相关函数的峰值没有图 5.8 那么明显地位于周期端点上、或基频不包括这些明显峰值时,又或是峰值很多而导致峰值检测复杂度增加的时候。此时,可再一次用 ZCR 算法中的移动平均滤波器来

解决这一问题,但并没有什么事是可以十全十美的,没有哪一种算法可以完全准确无误地检测所有乐器的所有音高。迄今为止,该算法不仅在技术上存在难点,而且也很难计算音高本身,因为准确地说,音高是一种声音的听觉感知,需要考虑心理声学的因素。

5.3 互相关函数

如式 5.10 所示,与自相关函数不同的是,互相关函数(cross-correlation)指的是两个不同信号之间而不是一个信号之中的相关性。这里, x_1 和 x_2 是两个不同类型的信号, N 是帧长, τ 是延迟, n 是采样点序号。

$$r_{x_1 x_2}[\tau] = \sum_{n=0}^{N-1} x_1[n] \cdot x_2[n + \tau] \quad (5.10)$$

与对同一波形进行自相关函数分析不同,可用互相关函数来计算两个不同类型信号之间的相关性,如可计算一个钢琴波形与一个小提琴波形的相关性。这样无论出于何种目的,都可用互相关函数来检测不同类型信号之间的相关性。又如可通过计算输入波形与硬盘中既有的 100 个波形数据库中信号的互相关函数,来获得相似度关系,根据互相关检测可粗略地对输入样本进行自动分类。

6. 采样率变换

采样率变换(sample rate conversion)是指通过增加或者减少采样点而改变信号采样频率的过程。增加采样点的过程称为过采样(up-sampling),减少采样点的过程则称为减采样(down-sampling)。本节将从时域的角度讨论如何增加和减少采样点,以及其他相关的问题,并在介绍傅立叶变换后,于第 8 章再次探讨该问题,并了解过采样或减采样给输出信号所带来的人工痕迹和失真。

6.1 过采样

过采样是指在信号中加入一系列幅值为 0 的采样点以增加信号样点的过程,定义如式 6.1,其中 L 为整数,是输入信号 x 的过采样量, n 为采样点序号。

$$y[n] = \begin{cases} x[n/L], & \text{当 } n/L \text{ 为整数时} \\ 0, & \text{当 } n/L \text{ 非整数时} \end{cases} \quad (6.1)$$

上式看起来或许有一点混淆,而代入数值其含义将变得清晰。如令 $x[n]=n$,其中 $n=0, 1, 2, 3, 4$,且 $L=2$,则有:

$$\begin{aligned} y[0] &= x[0/2] = x[0] = 0 \\ y[1] &= x[1/2] = 0 \\ y[2] &= x[2/2] = x[1] = 1 \\ y[3] &= x[3/2] = 0 \\ y[4] &= x[4/2] = x[2] = 2 \\ y[5] &= x[5/2] = 0 \\ y[6] &= x[6/2] = x[3] = 3 \\ y[7] &= x[7/2] = 0 \end{aligned} \quad (6.2)$$

由式 6.2 可知,由于 $L=2$,过采样后的输出信号长度是原信号 x 的两倍,零点与原信号样点相互交错,如图 6.1。

左图为原信号,中图为过采样后的信号。需要注意的是,过采样后的输出信号长度已经以 $L=2$ 的过采样量扩展为原信号样点数的两倍,并交错夹杂着零点。显然该信号不是平滑的波形,而如果过采样信号不经平滑处理会产生诸多问题。在第 8 章处理频域的人为失真时,将详细讨论相关问题。此时,需要使用一些平滑处理方法使过采样信号更接近于原信号,而这种平滑处理方法是通过对插值(interpolation)进行的。图 6.1 的右图给出了一个线性插值(插入值为相邻两采样点的平均幅值)后的 x 的过采样信号,由图可知线性插值后的过采样信号变得更加平滑,而整个过采样过程也完成了。

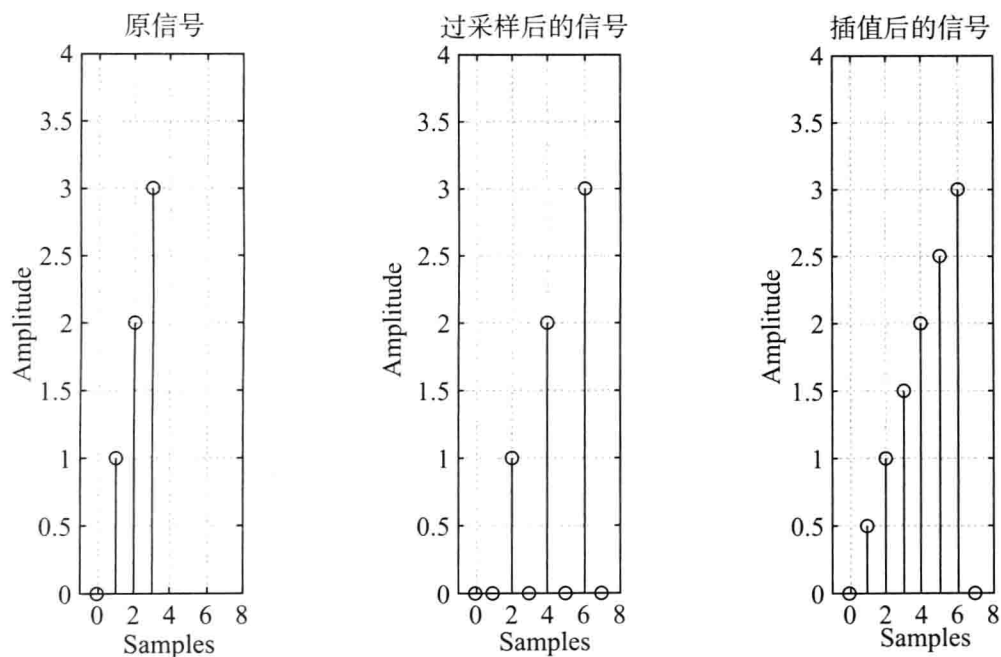


图 6.1 原信号(左),过采样(中)与过采样后的线性插值(右)

6.2 减采样

减采样与过采样相反,是更类似于时域及能量均连续的模拟信号的数字化。与采样类似,减采样后的信号是原信号的有限重构。即和模拟信号数字化一样,减采样后的信号长度将减少。 M 倍折叠减采样的定义如式 6.3,其中 M 为整数,是输入信号 x 的减采样量, n 为采样点序号。

$$y[n] = x[n \cdot M] \tag{6.3}$$

如令 $M=2$,减采样结果如图 6.2 所示。

如第 1 章介绍频率混叠与采样时所述,减采样在本质上也是对信号的限制(与对模拟信号进行采样类似),是否会发生频率混叠同样是需要考虑的问题。如果在减采样前没有对输入信号 x 的频率带宽进行合理地限制,将会出现频率混叠。在第 8 章将会讨论用反混叠滤波器(anti-aliasing filter)来处理无带宽限制的减采样过程。

在音乐应用中,采用过采样或减采样来调制采样率通常会获得有趣的音乐效果。如果保持声音输入设备不变,对信号的采样率进行调制,音高及频率特征的变化如表 6.1 所示。

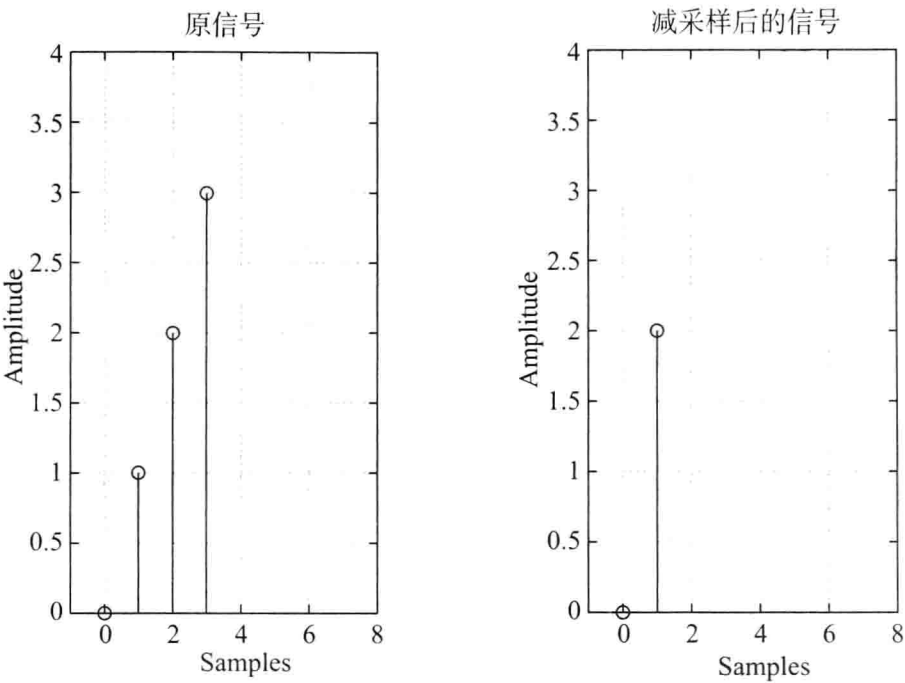


图 6.2 两倍折叠的减采样

表 6.1 声音信号的过采样和减采样与音高偏移的关系

	总采样点数	音高/频率变化
过采样	增加 L 值	偏低
减采样	减小 M 值	偏高

即使保持采样率 f_s 与音频输出设备一致,音高也会根据 L 和 M 的值上移或下移,由于总采样点数的增加(由 L 值决定)或减小(由 M 值决定),声音特征或者说它的音色也会发生变化。可用减采样来模拟高音移调的效果,而这种所谓“芒奇金效应”(munchkin effect)的影响通常是负面的,除非作品本身需要这样的效果。读者可用下述 MATLAB® 代码来播放一个用减采样来制造芒奇金效应的人声信号。相反的,过采样可用来在时域上扩展信号——如将信号长度扩展为两倍。由于在原信号中加入了采样点,音高也会发生与上述相似的偏移问题。

```
%Load some audio sample into MATLAB
[x, fs] = wavread('myInstrumentSound.wav');
sound(x, fs)

% try it with up-sampling, for example
sound(upsample(x, 2), fs)

%or down-sampling, for example
sound (downsample(x, 2), fs)
```

代码 6.1 减采样与过采样示例

第 9 章将介绍如何在保持信号长度不变的情况下改变音高的方法。下一节则将介绍一种

在保持音高不变的情况下,采用叠加 OLA (overlap-and-add) 算法对信号进行时域延展(变长)或时域压缩(变短)的简便方法。

7. 叠 加

采用过采样或减采样来增长或缩短信号时,不可避免地会产生上述的副作用。解决这一问题的方法之一称为叠加(OLA)。采用 OLA 叠加算法可以在保持音高不变的情况下,即不带入“附加”信号信息的情况下,对信号进行时域延展或时域压缩。这一令人惊讶的简便算法的核心思想如图 7.1 所示。

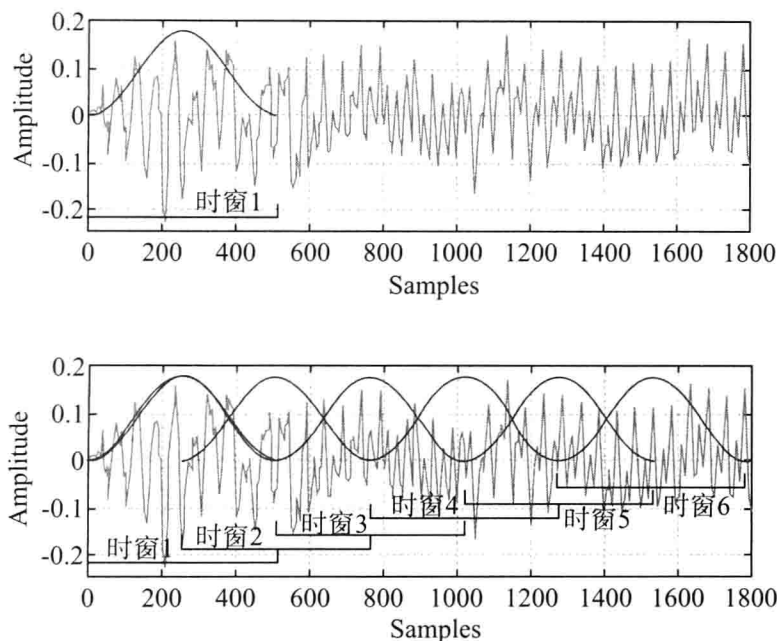


图 7.1 汉恩窗的叠加

为了学习该算法的技术细节,以先前钢琴声的前 1,800 个采样为例,图 7.1 的上图为经一个宽度由汉恩窗截取的该钢琴声片段。之前的时窗化实例中,时窗在截取片段时,下一个时窗的起始采样点恰位于前一个时窗结束部分的右侧边缘。而如图 7.1 的下图所示,我们注意到这 6 个时窗不像之前的例子那样非重叠地相继连续,而是进行了重叠设置。更准确地说,在此例中,用一个宽度为 512 个采样点(即 110 毫秒)的汉恩窗进行叠加,时窗叠加率为 50%,相当于 256 个采样点(即 5 毫秒)。每一个新时窗起始时间的间隔不再等于之前的时窗宽度,也不再恰好等于时窗宽度的整数倍,而是等于更小的叠加宽度的整数倍,这一叠加宽度称为跳距(hop size),此例跳距为 256 个采样点。如果有 90% 的部分进行叠加,则需要的跳距为 180 个采样点。图 7.2 为上述 6 个时窗截取的图 7.1 钢琴声的片段,长度均为 512 个采样点。若再用这 6 个时窗化后的片段以不同的叠加率首尾相接,则会获得一个更长或更短的信号。该信号的长度取决于合成阶段所选择的叠加率。其实也可用矩形窗来进行叠加,但因为汉恩窗的平滑边缘有助于在叠加信号融合相邻时窗的信号(交叉叠化),所以选择汉恩窗而不是矩形窗。OLA 叠加算法的第一步是分析信号即对原信号进行时窗分析,第二步实际上合成的新信号即是将时窗后的信号进行叠加。因此,该算法包含了对信号的分析 and 重构两个部分,总称为叠加。

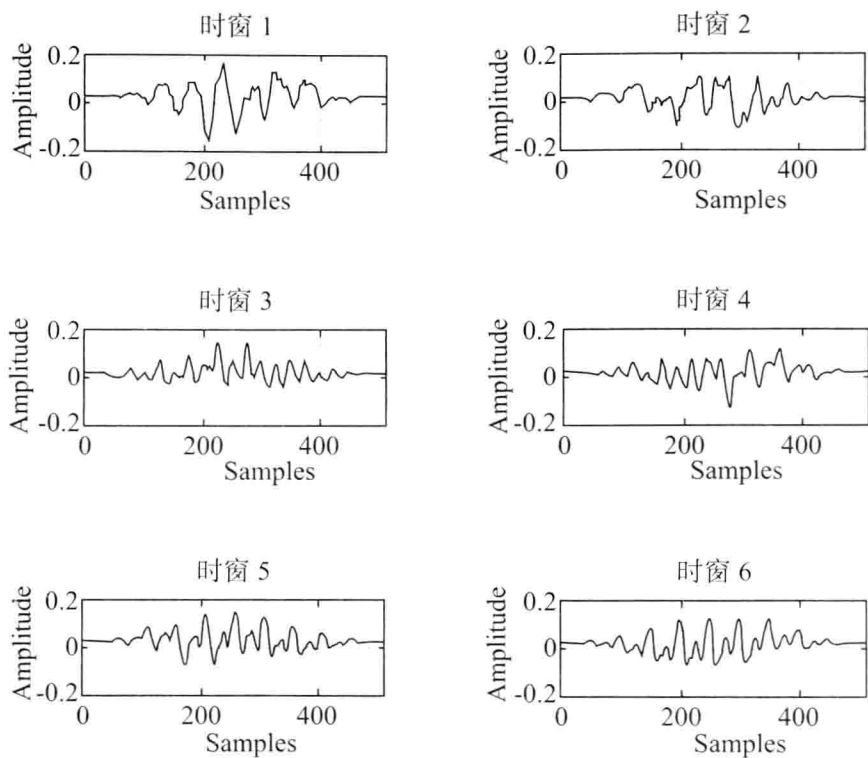


图 7.2 由时窗截取的钢琴声音片段

如果使用 50% 的分析叠加率(256 个采样点)分析信号,再运用 75% 的合成叠加率(384 个采样点)重构信号,则以 OLA 算法重构后的信号长度比原信号缩短了 25%,总共有 1,280 个采样点。例如如果使用 25% 的叠加率,则会得到一个时间延展信号,其长度比原信号长 25%,总共有 2,250 个采样点。有趣的是,在以上两种情况下,都不会感觉到音高有变化,因为 OLA 叠加算法没有对信息进行过采样或减采样。

OLA 叠加算法利用了心理声学与完形理论中有关的观点来处理对整体感知的倾向性。正如第 1 章在提到对音长的感知时曾述及,人耳是通过声音事件发生和持续的长短来作出判断的。当脉冲序列频率只有 1Hz 时,人耳并不能听到该声音的频率,而是只能数出脉冲的个数,如果每个脉冲的强度有区别的话就会形成节奏感。随着脉冲序列频率逐步加快,越来越难数清脉冲的个数,而整个声音的音色也会发生改变,直至到达 20Hz 附近。超过 20Hz 后,人耳开始把每秒内的脉冲当做一个声音来判断,渐渐便产生了音高的感觉。当频率再提升至接近人耳听觉感知频率上限时,音高感知将趋向于对更高频率的感知。OLA 叠加算法就是利用心理声学的这一特征,在 20~30Hz(对应时长为 30~50 毫秒)人耳频率感知模糊的临界频率内,加入一个宽度很小的时窗以及很小的跳距,来使信号长度及其时长感知发生变化,同时不改变音高。

叠加算法的局限及其解决办法

OLA 算法虽然简单有效,但需要进一步的调整来提高它工作的鲁棒性。该算法的一个常见问题是因采用周期间隔(即固定跳距)时窗而产生的——假设有足够证据可以证明的话,那么任何周期短至人耳能够听到的范围时,该声音就能够被感觉到。回到图 7.1,我们注意到由时窗叠加所导致的周期模式会让包络产生突起。这一问题与以上提到的心理声学理论有关——即对历时事件的听觉感知包括节奏、粗糙感和音高感觉。就是说,如果沿时窗宽度设置的跳距

足够短,由声波相消和相长干涉(详见第 4 章)所导致的人为失真本身就会带来节奏调制、粗糙感和音高感觉。该问题的一个解决方法是采用非周期跳距,即沿信号设置时窗叠加步进时采用微变化跳距或随机跳距。尽管该方法多少能缓解基于节奏或音高的人为失真,但这种调整会增加噪度而使输出信号听上去更加粗糙。

对音调声音进行时间延展和压缩时,另一种提高 OLA 叠加算法音质的方法称为音高同步叠加算法 PSOLA (pitch synchronous overlap-and-add),有时也称为时域谐波缩放算法 (time-domain harmonic scaling method)。该算法本身也相当简单,其做法是让每个时窗中心的间距与时窗化片段的基频保持一致和同步,如图 7.3 所示。因此每个时窗的中心位置都严格对应着基频的整数倍,或者用轻微的随机调整使每个时窗的中心不总是完全同步,以减弱粗糙感并在输出信号中强化基频因素。由于 PSOLA 算法以基频来掩盖时窗跳距的周期,因此由 OLA 叠加算法带来的人为失真也将减小。显然,为了提高 PSOLA 算法的准确性,需要一个良好的基频跟踪器,如果信号本身没有确定的音高,则 PSOLA 算法也不适用于此类信号。

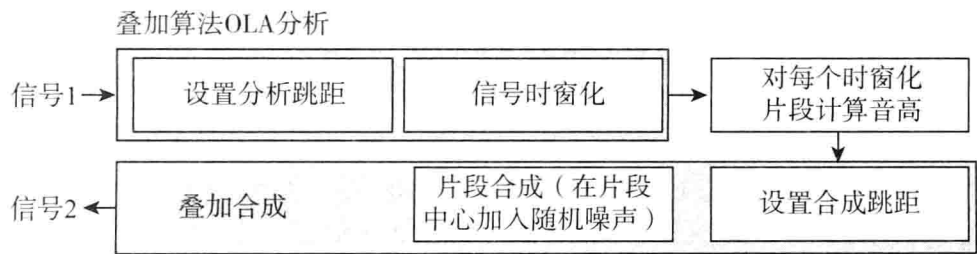


图 7.3 PSOLA 算法

8. 乐例分析

作为一个贝斯吉他手,笔者很欣赏具有传奇色彩的“皇后摇滚乐队”(Queen)在上世纪八十年代早期发行的歌曲《给尘土的另一个吻》(Another One Bites the Dust)。该曲随着人声的主体节奏,能明显感觉到伴奏中有一个固定的机械节奏型,这种节奏能将现场气氛推向狂热的顶点,这无疑是这首歌如此受欢迎的重要原因之一。令人感兴趣的是,皇后乐队采用了许多声音处理技术来制作这首歌曲,包括钢琴样本的反转回放,铙钹声,以及其他今日看来用一台台式电脑或笔记本电脑就可以完成的声音效果。而回到上世纪八十年代,这些效果是难以做到的,特别是该歌曲中的“鼓与贝斯循环效果”(drum and bass loop)。按流传说,当时只能事先录制鼓的一个节奏型,然后在现场演出中反复播放,就和现在的 CD 样碟一样。不管这种说法是否可靠,该歌曲的机械式节奏型对于当时的流行乐而言是一个先例。该歌曲另一个值得称道的地方是在波形中似乎有一些隐藏信息,也许将歌曲时间反转回放时会得到这些信息,即所谓的“倒带掩蔽”(back-masking)。我们将反转播放整首歌曲的机会留给读者,看看是否能找到这些或有或无的信号。此外,能否感觉到这些隐藏信息也许并不十分重要,有趣的是一支摇滚乐队竟利用电子声音调制技术来暗示乐曲是从后而起的,并鼓励听众倒退播放整首歌曲以获得更好的体验。

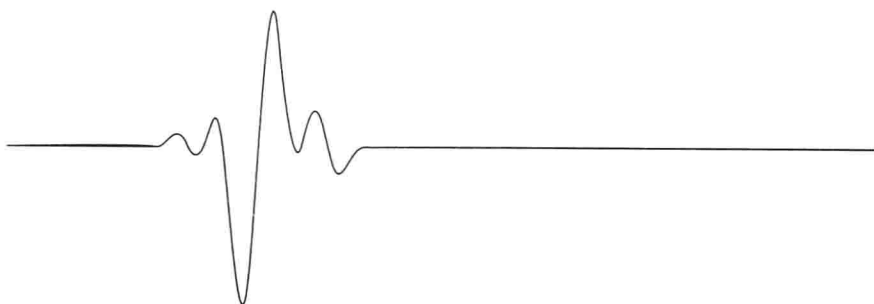
另一个乐于使用信号信息处理技术来获得循环效果的流行乐队是“电台司令乐队”(Radiohead)。尽管这种循环效果不是利用波表合成技术产生的,但电台司令乐队像许多能想到的过

去热门的乐队[如“平克·弗罗依德乐队”(Pink Floyd)和“钱乐队”(Money)]一样,以这种效果作为音乐的灵感。特别是现在随着计算机技术在音乐上的应用,截取以往经典作品片段进行再采样和循环处理也成为了音乐创作的灵感来源之一。如该乐队 2000 年发行的歌曲《傻瓜》(*Idiotique*)就是一个典型例子。这首歌曲之所以如此有魅力,是因为它没有采用经典流行歌曲的片段作为素材,而是采用了作曲家保罗·兰斯基(Paul Lansky)的作品《温和而轻柔地》(*Mild und Leise*)。这是一首完全通过计算机使用 FM 调频技术(详见第 4 章)和特殊滤波器来进行合成的曲子,乐曲合成代码又以肯·施泰格茨(Ken Steiglitz)的福特兰编程语言(Fortran IV)和乔治·珀尔(George Perle)的十二音体系为基础写成。《温和而轻柔地》显然应该归于电子及计算机音乐体裁,而不是传统音乐类型中的流行音乐。最令人惊讶的是,电台司令乐队这些家伙竟在第一时间找到这张 LP 唱片,并从中猎取灵感,最终将该曲作为《傻瓜》的基本动机,创作出了一首与 27 年前的这首原作几乎完全不同的全新作品,而只能在其循环音乐动机中找到原作的一些痕迹。说到电子音乐,另一首基于时域信号处理技术创作的成功作品是由作曲家威廉·斯克托尔德(William Schottstaedt)的《巨物》(*Leviathan*)。这首曲子在时域上以信号处理过程为基础,使用了来自于“效果唱片”的 14 个声源,包括火车、牛奶机、鸡、船的桅杆和水泵的声音。用于制作这个“带片作品”(tape piece)的材料,由软件程序剪辑成 15,000 个片段,然后以循环或拼贴的手法组合在一起。为了增加循环的多元性,作曲家通过调整循环点,使同一循环动机在听感上有了微妙的变化。作曲家对采样技术卓越的驾驭能力,让一个如此简单的技术成就了一件声音颇为美妙的艺术作品。

参考文献与扩展阅读

- [1] Cook R. Perry 2002. Real Sound Synthesis for Interactive Application. A. K Peters, Ltd.
- [2] Davis, G. and Jones R. 1989. The Sound Reinforcement Handbook. Milwaukee; Hal Leonard Corporation.
- [3] Park, T. H. 2000. “Salient Feature Extraction of Musical Instrument Signals”. Master's Thesis, Dartmouth College.
- [4] Rabiner, L. R. and Schafer R. W. 1978. Digital Processing of Speech Signals. Prentice-Hall, Englewood Cliffs, NJ.

第3章 时域信号处理(二)



1. 引言

第3章将继续介绍音乐信号的时域信号处理方法,并讨论一些经典的计算机音乐合成技术,包括粒子合成(granular synthesis)和波形整形合成(waveshaping synthesis)。同时本章也会介绍一些常见和广泛使用的数字信号处理效果,如动态压缩、动态扩展、失真、合声、镶边等在音乐家家用工作站、专业录音棚和吉他手效果包中常见的效果。本章的一些内容在后续章节中还会提到,以增强读者对信号处理概念和技术的理解。

2. 粒子合成

本书虽然由描述声信号的一般方法,即正弦波和声音信号的波动性开始,但声音也具有我们所熟悉的粒子性行为,即声音和光波一样会同时显现出粒像性(如光波中的光子)和波动性特征。粒子合成技术就是基于声音的粒子性理论展开的。有关用于粒子合成的声粒子(sonic grain)的记录也许最早见于荷兰科学家伊萨克·比克曼(Isaac Beekman)的日记。该日记被他保守了30年,直到他在1637年去世后才被公诸于世。后来因全息成像技术而获诺贝尔物理学奖的英国科学家丹尼斯·伽柏(Gabor 1947),在有关声量子理论的研究中进行了关于声粒子的论述。术语“粒子”一词后来又被希腊建筑师兼作曲家伊阿尼斯·克赛纳基斯(Iannis Xenakis)运用,他在该领域进行了大量研究,并创作了一系列基于声粒子理论的音乐作品。

和许多伟大的思想一样,粒子合成的基本概念其实很简单:即用时长较短的基本声粒子在时域上进行横向或纵向的并置,以合成一个新的声音事件或一片“声云”(sound cloud),这在某种程度上类似于电脑显示器用本身没有任何意义的像素(相当于粒子)的组合来获得清晰而生动的图像。如果可能,也可用上述图像的思想将声音理解为一幅声音景像——即适当定义声粒子密度和强度等参数后,就可以使用一些较小的声粒子来合成声音。粒子模型与像素模型不同的是,粒子的排列通常不如像素那样稳定而有序,而是在“声布”(sound canvas)上随机分布的,这将在下节进一步论述。粒子合成的工作原理利用了人耳听觉的局限性,以及与人耳声音感知有关的心理声学特征。其实,粒子合成在其核心思想上与第2章所讨论的OLA叠加算法

非常相似,主要差别是在粒子合成的时窗化信号(即声粒子)的纵向排列上。这一纵向排列重构成了给定时间点上的粒子密度,这些也将在后续的章节中详细介绍。

2.1 粒子合成的基本参数

本节将介绍粒子合成技术中的主要参数和概念。表 2.1 给出了这些典型的运用于粒子合成的基本参数。

表 2.1 粒子合成的基本参数

参 数	描 述
粒子长度(Grain size)	粒子的时长,典型的单位为毫秒。
窗类型(Windows type)	用于截取粒子的时窗类型。
粒子振幅(Grain amplitude)	粒子的最大幅值。
粒子密度(Grain density)	每秒内粒子的数量。
粒子音高(Grain pitch)	如果存在,指粒子的音高特征。

首先定义粒子合成中最为重要的概念——声粒子,它的定义为足够短的、由时窗截取的典型声信号片段,时长通常在几毫秒至 100 毫秒之间。时窗宽度的选择直接与所需要的声音效果有关,包括对原信号的解析度——时窗宽度直接影响单个声粒子的听觉感知特征。粒子长度越小,则在听一个独立的声粒子时,原信号的原始特征越模糊。伽柏的实验在检测声粒子的长度时发现,原信号的频率信息或许扮演着微妙的角色——伽柏指出声粒子的长度能反映出取决于声粒子本身频率的特征信息(Gabor 1946),即重构原信号的声粒子长度的分辨阈限是原信号频率成分(frequency composition)的函数。信号的频率成分是指重构声音事件所需的一系列具有特定振幅、频率和相位的正弦波,这一理论将在第 8 章讨论。

为了获得一个声粒子,需要为目标信号的典型片段选择一个时窗(时窗分析已在第 2 章第 3 节中讨论)。将声粒子定义为由时窗 $w[n-L]$ 所截取的输入信号 $x[n]$ 的时窗化片段,则其计算如式 2.1,其中 L 为时延,即时窗的起始位置或偏移量。

$$x_{grain}[n] = w[n - L] \cdot x[n]$$

(2.1)

常见的获得声粒子的时窗类型包括如图 2.1 所示的汉恩窗与三角窗(时窗类型的详细讨论见第 8 章)。以上两种时窗所显示的向两侧衰减的平滑边缘有利于合成过程中声粒子的混叠与融合。

若令输入信号 $x[n]$ 中 $n=0\sim 4\,999$, $L=0$,时窗宽度 $N=100$,则从输入信号的起始点可获得相当于输入信号的时窗化片段的声粒子 $x[0], x[1], \cdots x[98], x[99]$ 。而当 $L=100$ 且 $N=100$ 时,将同样的时窗 w 截取出原输入信号的片段 $x[100], x[101], \cdots x[198], x[199]$ 。

粒子振幅可用如式 2.2 所示的缩放因子进行控制。如果将这些独立的粒子整理成合适的集合,则把它们适当地沿时间轴排列时,粒子的振幅就可以组成所需要的振幅包络。也就是可方便地通过调整每个独立粒子的振幅来控制整个信号的振幅特征。这种情况与计算机屏幕上的像素类似,即通过增加该区域中一组像素的亮度来提高显示器某一个区域的亮度。

$$x'_{grain}[n] = \alpha \cdot x_{grain}[n]$$

(2.2)

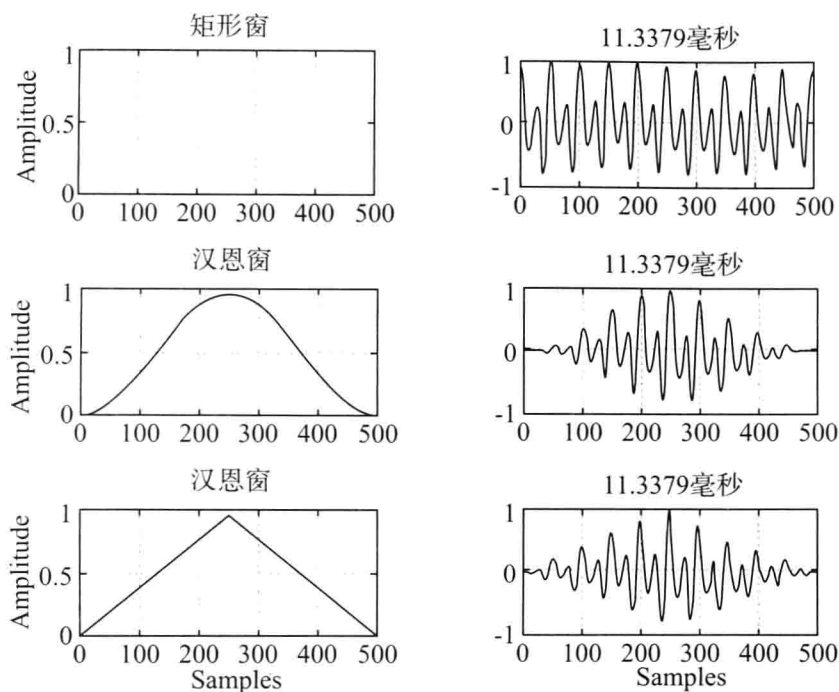


图 2.1 由矩形窗、汉恩窗与三角窗所截取的声粒子

粒子密度一般定义为每秒内粒子的数量,通常表征合成信号的丰满度。粒子密度越大,则声音也越丰满。如同控制每个粒子的振幅一样,也可用过采样和减采样技术来控制每个粒子的音高,进一步的讨论见 2.4 节。

2.2 异步粒子合成

本书将介绍一种常见的粒子合成类型,即异步粒子合成(asynchronous granular synthesis),其他的类型还有准同步粒子合成(quasi-synchronous granular synthesis)和音高同步粒子合成(pitch-synchronous granular synthesis)。异步合成的主要特征是,在时间轴上对粒子进行排列时,它们的分布是随机的,如图 2.2 所示。这样一来,尽管保持每秒内粒子的总数量(即粒子密度)不变,每个粒子的起始时刻位置也没有固定的模式或周期性。

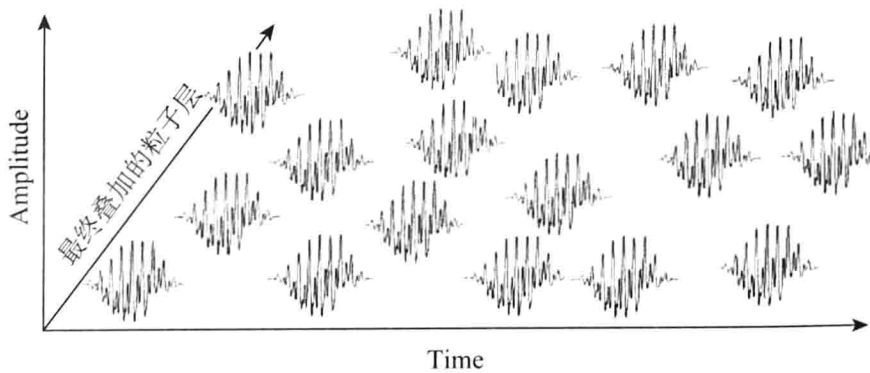


图 2.2 粒子的异步叠加和分层排列

图 2.3 的上图给出了一个 1 秒钟钢琴声的粒子合成实例,来自于钢琴声源的粒子长度为 500 个采样点(11.34 ms),时窗为汉恩窗,粒子密度为 20,000 个/秒(grains/sec),指数衰减振幅包络,采样率为 44.1 kHz。粒子在 1 秒钟内呈随机分布,并加和为合成后的输出信号。其中,图 2.3 的下图为合成信号的前 2,000 个采样点,显示出了图 2.1 中样的原始粒子的波形特征。

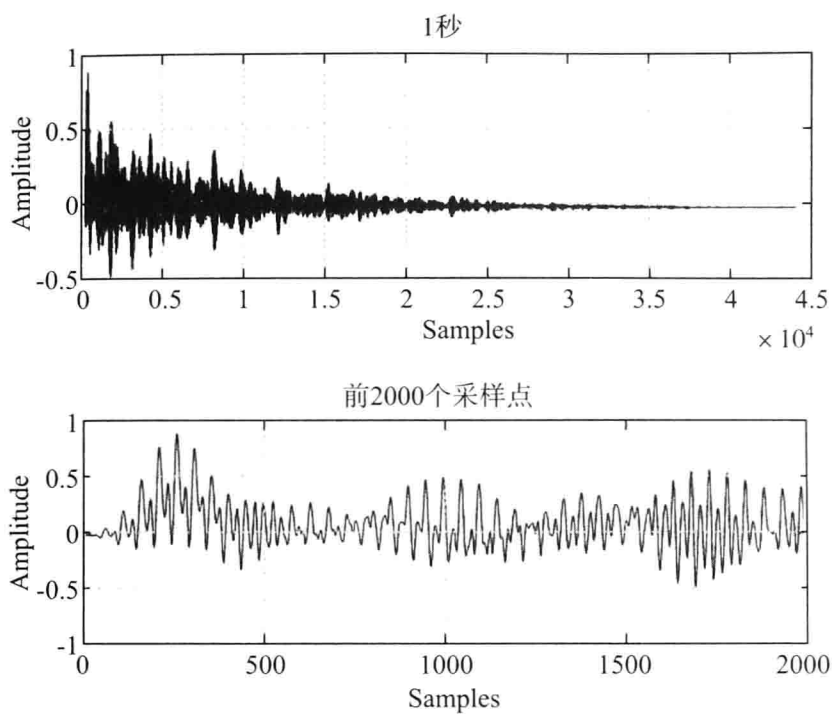


图 2.3 异步粒子合成

2.3 移调与时间延展及压缩

既然可对基本声粒子进行调整,那么就在粒子合成中直观地修改音高,并进行时间延展和压缩。为了改变音高并获得需要的音高,需要做的是进行过采样或减采样,适用于该过程的反混叠滤波器(anti-aliasing filter)将在第 7 章讨论。而用粒子合成来改变时长也同样简单——只需用粒子来填充所需的时长。例如,如果原声音事件的时长只有 1 秒,想要将其时间延展为 3 秒,需要做的只是用时窗截取的粒子以合适的粒子密度对这 3 秒钟的时间进行填充。当然,为了使合成后的音色自然,需要对信号做进一步的调整,包括按要求从原信号的激励、衰减、稳态

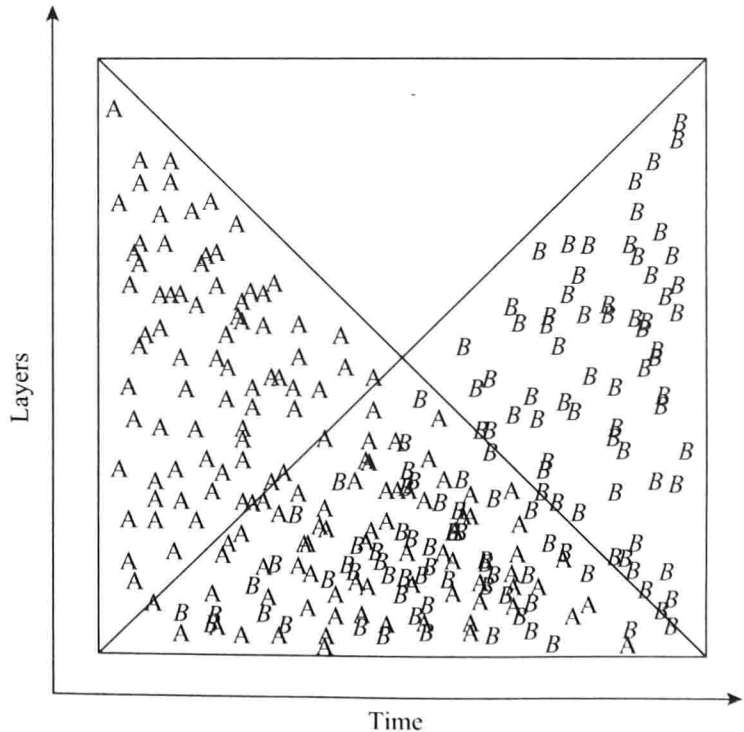


图 2.4 从音色 A 至 B 的声变形

和释音阶段中截取不同类型的粒子——原声音信号字节中的这些阶段往往有着不同的音色特质。

2.4 用粒子合成进行声变形

也可以用粒子合成在不同类型的声音事件之间进行声变形,或者用来自两个或更多不同音频信号的粒子合成混合声音事件。例如,如果目标是由音色 A 平滑变形至音色 B,则可从来自声源 A 的粒子(比例为 100%)开始,然后通过 A 粒子逐渐减少而同时 B 粒子比例增加的过渡区到达仅留下 B 粒子(比例为 100%)的区域,如图 2.4 所示。在此例中,两种声源类型的交叉过渡是线性的,也可按需要将其调整为指数或其他非线性曲线。

3. 振幅失真与波形整形

第 3 节将介绍一些在时域上控制和修改信号幅值(动态)的调制技术。在某种程度上,这些过程可看做是对输入波形的调整,因此进入到了波形整形技术(waveshaping)的一般范畴。波形整形通常是一种基于传递函数(transfer function)来对波形进行整形的时域声音调制方法。该方法的核心思想如图 3.1——输入信号的幅值被一个往往为非线性的传递函数进行整形和调制(线性将在第 5 章详细讨论)。

如果传递函数是线性的,可如图 3.2 表示为 $y=a \cdot x$,其中 a 为直线斜率的缩放因子。由于

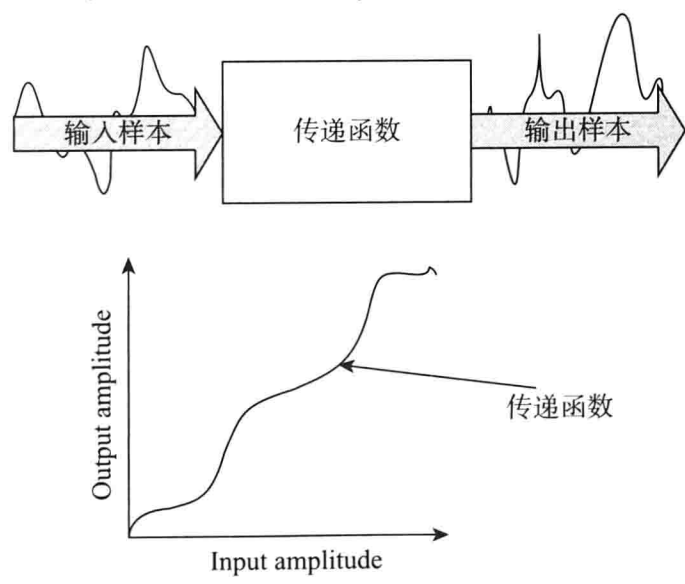


图 3.1 波形整形合成

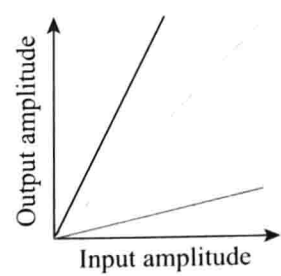


图 3.2 线性传递函数

该方法在本质上是通过对输入信号的振幅进行失真处理来获得需要的输出信号,类似于下节将要简要介绍的动态压缩,因此基于波形整形的技术又称为失真合成(distortion synthesis)。波形整形合成如此让人感兴趣的原因是输入信号幅值是时间的函数,则波形整形器(waveshaper)的输出信号也是动态变化的而不是静止的。因此,设计和计算传递函数以获得目标信号和音色就成为了波形整形合成的核心。

3.1 动态压缩

动态压缩器(dynamic compressor)常用于动态范围较大的声音,如电贝司或者由底鼓、小军鼓和嗵嗵鼓所组成的鼓组等。压缩器能够动态地减小输入信号的振幅,从而有效地以非线性的方式减小信号的动态范围。图 3.3 描述了该概念背后的一个简单压缩器的基本工作原理。

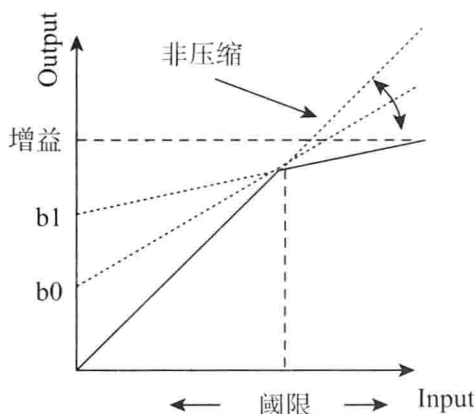


图 3.3 动态压缩器

压缩器的主要效果是降低选定范围内的输入信号的振幅——如图 3.3 所示的输入信号中的较高幅值。其结果是,在“低幅值”阈限与“高幅值”之间的幅值降低了,从而由宽至窄降低了动态范围。动态压缩的核心思想是在降低“高幅值”的同时保持“低幅值”不变。宽的动态范围能使乐器的声音变得非常柔和或非常响亮——最柔和与最响亮的声级差越大,动态范围就越宽。鼓组的运用和电贝司演奏中的“拍击”(slap and pop)技法是获得较宽动态范围和较好音质的例子。压缩器的压缩量主要取决于两个参数——阈值和波形整形器的传递函数。如前所述,传递函数表征了波形整形器的曲线形状,决定了输入、输出信号之间的关系,如图 3.3。而振幅阈值沿输入轴移动(图 3.3 的 x 轴)决定了什么范围以上的输入幅值被压缩。阈值越接近于 0,压缩区域越大。此例中,第 2 个参数是仿射线性函数(affine linear function) $y = mx + b$ 。由图 3.3 可知,在 0 到阈值的区域内,输出信号幅值 y 等于输入信号幅值 x ,而幅值超出阈值时,波形整形器以 $y = mx + b$ 的方式输出信号幅值。当 $b = 0$ 时,该仿射线性函数变成了线性函数(即无压缩, $y = x$)。压缩级则受步进斜率(m)的控制——斜率越小,压缩量越大,反之亦然。显然这里介绍的是一个非常粗略的例子,而商用压缩器在阈值点处有一个平滑的过渡区,称为“软转介”(soft knees)。不过,一般压缩器的效果都与上述例子类似。下面列出了上述动态压缩器的 MATLAB® 代码的主要部分。

上述代码首先记录输入信号幅值的正负号(极性),然后计算其绝对值,以确定哪一部分幅值将会被压缩,最后在合成输出幅值时再用 $|x|$ 乘以原信号的正负号。

用于进一步优化和调整压缩器的另外两个重要参数是压缩器的激励时间(attack time)和释


```
theSign = sign(x); % retain sign of waveform
x = abs(x); % get absolute value of x
for i=1:length(x)
    if x(i) > threshold
        % compress
        y(i) = (slope*x(i) + intercept)*theSign(i);
    else
        % do not compress
        y(i) = x(i)*theSign(i);
    end
end
end
```

代码 3.1 简单的动态压缩器

放时间(release time)。激励时间和释放时间是指压缩器开始作用(激励)的速度有多快,以及结束作用(释放)并回到低于阈值区域的速度有多快。因此,激励和释放时间本质上控制着非压缩声音与压缩声音之间的过渡时间,反之亦然。如果没有这两个重要参数,未经处理的非压缩部分和经处理输出的压缩部分之间的过渡会变得过于突然,从而产生微小而无意义存在的人为失真,问题的严重程度决定于输入信号的类型。图 3.4 描述了激励时间、释放时间和延迟的含义,压缩器在激励时间内开始工作,并在释放时间内结束工作。

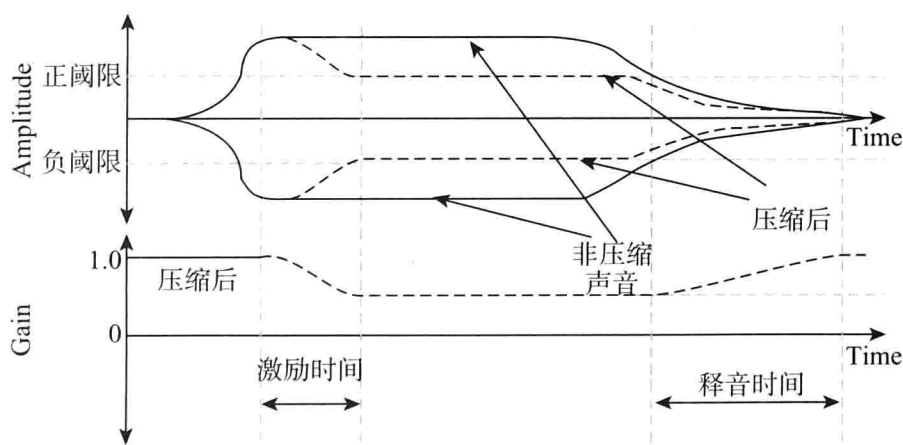


图 3.4 压缩器的激励时间与释放时间

压缩器的另一种形式是“限制器”(limiter),这也是一个音频应用软件中常见的效果器。限制器通过对信号的“硬限幅”(hard-limiting),有效保证输出信号(如通过扬声器输出)不超过设定的振幅声级。因此,限制器经常用于因传声器的存在而使声音反馈成为重大问题的现场演出或音乐会中。

3.2 失真

第一章已介绍过的削波失真在极端情况下会使波形变成方波。而此时了解压缩器的工作原理后,也可将失真理解为一个极端的动态压缩器——即当 $m = 0$ 时 $y = mx + b$ 变成了一条直线。失真和压缩的主要区别在于它们的应用场合不同。对于失真效果,由于高增益输入所带来的严重削波会产生额外的谐波,从而可按需要获得更具丰满度的声音。对于压缩器,这种削波

特征一般是不需要的,削波会降低输出信号的动态范围而在理想状态下不影响输出信号的音色。而对于失真效果器,同样的人为失真会成为所需要的音乐元素,从而产生音色调制效果。

3.3 动态扩展

和压缩器相反的是扩展器(expander)。与压缩器降低较高幅值的增益不同,它通过衰减较低幅值的增益来增加信号的动态范围。与压缩器一样,扩展器也可以看作是一个特殊的波形整形器。扩展器的幅值特征曲线如图 3.5 所示。

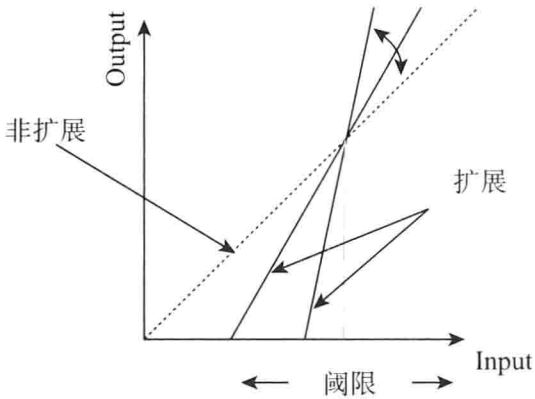


图 3.5 动态扩展器

与压缩器类似, $y=mx + b$ 的斜率和阈限决定了扩展量。当 $y=x$ 时,没有扩展,输出信号与输入信号增益相同。令人感兴趣的是,用于降低信号噪音的经典效果器“噪音门”(noise gate)的工作原理与扩展器相同。它们之间的不同之处在于,对于噪音门,低于门限阈值的输入幅值降至 0,而只输出高于门限阈值的输入幅值。用这种噪音门进行“降噪”时,只输出足够“强”或足够“亮”的非零幅值。而这种效果不能被看做是真正的降噪,因为实际上对于噪声本身它没有做任何处理——对于一个幅值在噪音门阈限(通常是很低的值)以上的输入信号,噪音会被较强部分的信号所掩盖,如高信噪比的小军鼓。不过噪音门对于有传声器的录音棚或现场演出是非常有用的。例如,当主唱不是对着麦克风演唱而是在舞台上走来走去时,在没有设置噪音门的情况下是很危险的,此时通过该话筒拾取的其他声音也会被放大并通过声系统播放出来,同时由于没有主唱声音的掩蔽,这些声音很可能被观众听到或者形成声反馈。而如果使用了噪音门,功率放大器就只能输出使用中的传声器的信号——即主唱对着麦克风喊叫的声音。

3.4 磁带饱和度失真

在以前只有磁带或磁带机的时候,录音棚只能使用模拟录音技术——这些技术由于磁带本身和磁带机技术的缺陷往往会带来非线性失真。其中一种重要的失真类型称为“磁带饱和度失真”(tape saturation)。简单地说,磁带饱和度失真是一种平滑而轻微的一类压缩人为失真,它是伴随着深受摇滚(或流行)音乐家们喜爱的模拟磁带机的出现而发展的。一种对磁带机所带来的“误差”进行数字仿真的方法是对音频信号进行平滑而细微的动态压缩。如第 1 节的例子所示,信号削波会产生谐波失真,而对输入信号进行压缩处理也会产生轻微的人为谐波失真,有时候这种失真对于某些音乐场合是不需要的。尽管很难说人们能够消除由于使用模拟磁带录音技术而带来的这些来自磁带机的人为失真,但数字系统既允许灵活地模仿这种相对非规则的效果,也进一步在它对所寻求的音乐效果没有帮助时选择不使用这种技术。而磁带一般不会给我

们带来这样的选择。

3.5 波形整形合成

波形整形合成的原理本质上与动态压缩是一样的,只不过可以按需要选择或设计,来传递函数和加入到波形整形器中的输入信号。波形整形合成的目标是选择传递函数和输入信号,通过细致和精确地控制来生成所需要的复杂谐波。本小节将介绍如何设计合适的传递函数。

3.5.1 第一类切比雪夫多项式

一种经典而常用的设计传递函数的方法是构造第一类切比雪夫多项式(Chebyshev polynomial of the 1st kind)而不是第三类近似多项式(Chebyshev polynomial of the 3rd Kind)。众所周知,多项式的一般形式如式 3.1。

$$f(x) = a_N \cdot x^{N-1} + a_{N-1} \cdot x^{N-2} + \cdots + a_1 \cdot x + a_0 \quad (3.1)$$

切比雪夫多项式的特点是,如果波形整形器的输入信号是一个正余弦信号,它可以产生特定的谐波。使用多项式的优点很大程度上在于其产生特定数量谐波的能力,并且因只使用一个独立的振荡器而具有很高的计算效率。采用切比雪夫多项式生成特定谐波结构的技术也被称为“频谱匹配”(spectral matching),它被广泛应用于匹配和模仿声学乐器的谐波结构中。当输入 x 是一个正余弦波时,任何基频的谐波都可用式 3.2 的多项式来描述,其中 T 为谐波分量,整数 k 表示谐波阶数,当 $k=1$ 时,表示该分量为基频, $k=2$ 时该分量是基频的八度泛音,则谐波 T_{k+1} 为基频的 $k+1$ 次谐波,以此类推。

$$T_{k+1}(x) = 2 \cdot x \cdot T_k(x) - T_{k-1}(x) \quad (3.2)$$

上述关系式仅对于输入信号 x 为正余弦波时适用。使用第一类切比雪夫多项式时,仅用一个给定基频而不需要附加的振荡器,就可在基本正余弦波中加入谐波,这使它成为了一种通过单频正余弦波高效合成的复合音,即由一个正余弦量生成一组正余弦量的算法。为了理解该类多项式生成特定谐波结构的原理,首先从以下一些基本三角公式开始了解第一类切比雪夫多项式。

$$\cos(a+b) = \cos(a) \cdot \cos(b) - \sin(a) \cdot \sin(b) \quad (3.3)$$

$$\sin(a+b) = \sin(a) \cdot \cos(b) + \cos(a) \cdot \sin(b) \quad (3.4)$$

由上述基本三角公式可推导出以下关系:

$$\begin{aligned} \cos(2 \cdot \theta) &= \cos(\theta + \theta) \\ &= \cos(\theta) \cdot \cos(\theta) - \sin(\theta) \cdot \sin(\theta) \\ &= \cos^2(\theta) - \sin^2(\theta) \\ &= \cos^2(\theta) - \{1 - \cos^2(\theta)\} \\ &= 2 \cdot \cos^2(\theta) - 1 \end{aligned} \quad (3.5)$$

$$\begin{aligned} \cos(3 \cdot \theta) &= \cos(\theta) \cdot \cos(2 \cdot \theta) - \sin(\theta) \cdot \sin(2 \cdot \theta) \\ &= \cos(\theta) \cdot \{2 \cdot \cos^2(\theta) - 1\} - \sin(\theta) \cdot \{2 \cdot \sin(\theta) \cdot \cos(\theta)\} \end{aligned}$$

$$\begin{aligned}
&= 2 \cdot \cos^3(\theta) - \cos(\theta) - 2 \cdot \sin^2(\theta) \cdot \cos(\theta) \\
&= 2 \cdot \cos^3(\theta) - \cos(\theta) - 2 \cdot \{1 - \cos^2(\theta)\} \cdot \cos(\theta) \\
&= 2 \cdot \cos^3(\theta) - \cos(\theta) - 2 \cdot \cos(\theta) + 2 \cdot \cos^3(\theta) \\
&= 4 \cdot \cos^3(\theta) - 3 \cdot \cos(\theta)
\end{aligned} \tag{3.6}$$

由以上三角公式类推可计算 $\cos(2 \cdot \theta)$ 和 $\cos(3 \cdot \theta)$, 进一步可计算随后的余弦形式 $\cos(m \cdot \theta)$, 其中 m 为整数。前五个余弦函数值如下:

$$\cos(0 \cdot \theta) = 1 \tag{3.7}$$

$$\cos(1 \cdot \theta) = \cos(\theta) \tag{3.8}$$

$$\cos(2 \cdot \theta) = 2 \cdot \cos^2(\theta) - 1 \tag{3.9}$$

$$\cos(3 \cdot \theta) = 4 \cdot \cos^3(\theta) - 3 \cdot \cos(\theta) \tag{3.10}$$

$$\cos(4 \cdot \theta) = 8 \cdot \cos^4(\theta) - 8 \cdot \cos^2(\theta) + 1 \tag{3.11}$$

此时, 令 $x = \cos\theta$, 引入变量则定义 $T_k(x)$ 可得式 3.12, 其中 k 为整数。

$$T_k(x) = \cos(k \cdot \theta) \tag{3.12}$$

以新定义的 $T_k(x)$ 改写式 3.7~3.11, 则有:

$$T_0(x) = \cos(0 \cdot \theta) = 1 \tag{3.13}$$

$$T_1(x) = \cos(1 \cdot \theta) = \cos(\theta) = x \tag{3.14}$$

$$T_2(x) = \cos(2 \cdot \theta) = 2 \cdot x^2 - 1 \tag{3.15}$$

$$T_3(x) = 4 \cdot x^3 - 3 \cdot x \tag{3.16}$$

$$T_4(x) = 8 \cdot x^4 - 8 \cdot x^2 + 1 \tag{3.17}$$

$$T_5(x) = 16 \cdot x^5 - 20 \cdot x^3 + 5 \cdot x \tag{3.18}$$

.....

此时观察到方程出现了模式化, 上述多项式的展开可归纳为如式 3.19 的一般形式, 即式 3.2 一开始的形式。

$$T_{k+1}(x) = 2 \cdot x \cdot T_k(x) - T_{k-1}(x) \tag{3.19}$$

上式就是第一类切比雪夫多项式。它之所以如此有用的原因是, 如果如上所述设输入 $x = \cos(\theta)$, 即余弦波的振幅为 1, 则每个切比雪夫多项式能够形成一个特定的波形整形器, 将只输出一个特定谐波。也就是说, 当 $x = \cos(\theta)$ 时, 切比雪夫多项式为 $T_k(x) = \cos(k \cdot \theta)$ 会产生一组余弦信号, 而不需要额外的振荡器, 这样可以减轻计算机处理器的负担。因此, 如果只要将一系列不同的 $T_k(x)$ 分量进行组合, 就可重构如小号谐波结构, 即不需要使用数量相同的振荡器就可通过波形整形合成, 用大量的谐波生成一个合成的小号声。例如, 波形整形器根据式 3.20 可合成一个类似方波的声音。

$$F(x) = T_1(x) + \frac{1}{3}T_3(x) + \frac{1}{5}T_5(x) + \frac{1}{7}T_7(x) + \frac{1}{9}T_9(x) + \frac{1}{11}T_{11}(x) \tag{3.20}$$

如前所述,这种用于波形整形的方法称为频谱匹配——即传递函数可与既有的声学乐器或其他任何声音事件的谐波结构相匹配。这种方法有效而盛行,它之所以能成为强大的声音合成工具的原因是,它不仅不需要价格昂贵的振荡器,而且当输入非余弦波时还有着令人感兴趣的音色输出结果。同时它不仅用余弦波进行输入。

此外,需要注意的是,当输入的是数字信号时,必须对上述切比雪夫多项式所构造的信号频率加以限制,因为实际不需要无限的带宽来构造该多项式。换言之,必须确保所产生的来自于基频的谐波分量(取决于 k)符合奈奎斯特采样定律的限制,即:

$$k \cdot f_0 < \frac{f_s}{2} \quad (3.21)$$

对于由许多无法预知的频率或余弦波组成的复杂信号,应该对它的带宽有所限制,以使其满足切比雪夫多项式 $T_k(x)$ 的要求。这种限制技术称为低通滤波,将在第7章介绍。为创作音乐而使用何种技术没有对错之分,只要它最终能为作曲本身服务。如果结果令人满意,则作曲家或声音设计师使用了何种技术其实并不重要,就像有时候也制造频率混叠效果一样。

4. 几个相似的时域 DSP 效果

本节将介绍一组我们熟知的数字信号处理器,至少我们曾经听到过它们的效果。

4.1 等功率平移

“平移”(panning)一词在音频技术中无处不在,但实际上它来自于电影制作术语,是指用摄像机转动拍摄以获得某一场面的全景。音频技术中,平移一词来源于“平移旋钮”(pan pot),即“全景电位器”(panoramic potententionmeter)的缩写。电位器是一个可变电阻,类似于起居室中立体声音响设备或调光器上的音量或亮度控制旋钮。两通道音频信号的平移旋钮可将信号分配到两个输出通道和两个扬声器上。想象该旋钮向逆时针或顺时针方向转动,可使声音在左右两个扬声器的极端位置之间移动,如果将旋钮置于12点位置上,使两个扬声器的声音响度一样时,声音就位于两个扬声器连线的正中央。

这一思想在数字系统中同样可以实现。设信号的最大输出功率是一个浮点数1.0,通过两个扬声器平移至两个声道中, $x[n]$ 为输入信号, $y_{\text{left}}[n]$ 和 $y_{\text{right}}[n]$ 分别是左侧和右侧声道的输出信号,则有:

$$y[n]_{\text{left}} = x[n] \quad (4.1)$$

$$y[n]_{\text{right}} = 1.0 - x[n] \quad (4.2)$$

虽然由上式可知这一原理易于使用,不过这里却忽略了一个小问题,即人耳听觉系统的特性以及如何感知这种动态电平的变化。回顾第1章中所描述的人耳对声音强度的感知可知,人耳感知的声音能量应该是振幅的平方。换句话说,对于人耳,如果旋钮的旋转角速度是恒定的,则上述平移算法不可能将信号强度平滑而均匀地等分至左右两个声道。为了使平移效果接近听觉上的线性特征,可取信号能量的平方根值进行平移。显然平方根并不是一个线性函数,但重要的是,当用平方根来抵消振幅的平方时,能使平移旋钮的线性转动变化接近于人耳对声音

强度变化感知的线性特征。改进后的“线性感知平移方程”(perceptually linear panning equation)可改写为:

$$y[n]_{\text{left}} = \sqrt{x[n]} \tag{4.3}$$

和

$$y[n]_{\text{right}} = \sqrt{1.0 - x[n]} \tag{4.4}$$

通常听到的平移 -3 dB,是指当平移旋钮位于中央位置时,左右声道的电平相对于旋钮位于极左或极右位置时下降 3 dB,即等于 $20 \cdot \log_{10}(0.707)$,或者说是振幅的平方根值为 0.707,其中幅值 0.5 的平方根值为 0.707,而 0.5 位于 0.0 和 1.0 当中,即左右声道等分总幅值。图 4.1 给出了线性幅值与分贝幅值下的等功率平移过程。

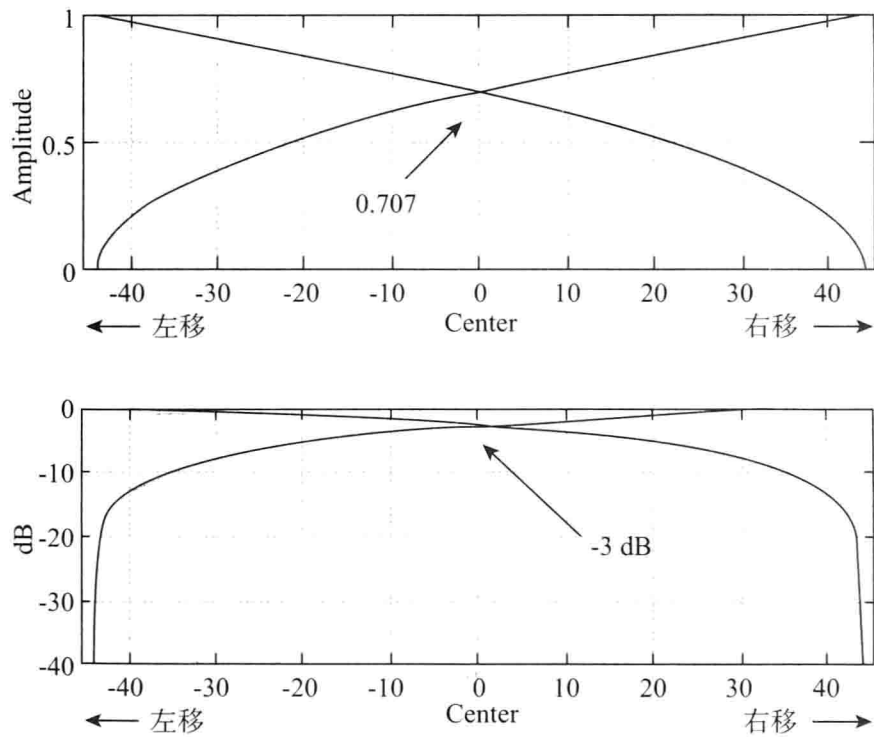


图 4.1 线性与对数振幅下的等功率平移

4.2 延迟

本章讨论的最后三种数字信号处理效果都是基于对输入信号进行不同程度的延迟。通过改变延迟量,再将延迟后的信号与原信号混合就能产生许多日常音频应用软件中所使用的有趣的音乐效果。而差分方程(difference equation)、延迟器(delays)和分数延迟器(fractional delays)将在第 5 章和第 7 章讨论。

4.2.1 回声、合声与镶边效果(声染色)

第一个以时延为基础的效果器是所谓的“回声”(echo),实际上在 20 世纪 80 年代,真的有一支乐队叫做“回声与兔子男人”(Echo and the Bunnymen),不过他们的乐曲并没有太多地使用这一效果。毫无疑问,在一个宽敞的大厅、大教堂、大峡谷或宴会厅内,高强度的叫喊声会以声速经界面反弹回来而产生回声效果。回声可用式 4.5 表示。

$$y[n] = x[n] + b_N \cdot x[n - N] \quad (4.5)$$

N 为延迟, 尽管对延迟长度没有严格限制, 而长延迟意味着长延时回声, 但通常经界面反射回来的时间为 10 毫秒至 50 毫秒。 b_N 是经界面反射后信号的衰减系数或缩放因子, 由于所有经界面反射后的声音有能量损失, 因此反射声的强度会轻微下降。

延迟器的时延通常会比回声短, 回声通常指合声 (chorus) 延迟。而回声与合声的差别在于合声效果能动态地改变延迟。延迟时间在 10 毫秒至 25 毫秒平滑变化时会获得“倍增效果” (doubling effect)。倍增后的声音就好像两个人合唱或两个音频信号叠加一样。随着系统中延迟器的增加, 能获得类似多人同时合唱的效果——因而取名为“合声效果” (chorus effect)。当众人按同一音高合唱或合奏同一乐器时, 往往会出现基频失调或轻微变化的情况。这是由于演唱者或演奏者的能力有差异, 或者是乐器本身 (如小提琴, 人声和圆号等) 的音准存在微小差异, 又或者是因听众与每个声源有声程差而造成的。即使每件乐器的音准都调得很好, 但随着声源数目的增加, 多多少少也会出现基频失调的现象。数字合声算法可用可变延迟器来模拟这种缺陷, 以使声音听起来更厚、更密实:

$$y[n] = x[n] + x[n - g(n)] \quad (4.6)$$

$g[n]$ 是一个整数延迟器, 它随时延的上下限缓慢变化。例如可以把 $g[n]$ 设计为一个低频数字正弦振荡器, 即低频振荡器 LFO (low frequency oscillator)。

早在模拟磁带录音机盛行的年代, 录音师有时候会用两台独立的磁带机来录制两个内容相同的音轨。但由于数字技术的出现, 模拟磁带录音机虽然还在一些录音棚中使用, 但数量已经大为减少。如果以手压法兰 (RIM) 的方式轻微改变磁带轮的速度以使播放速度产生微小偏差, 即使会使输出信号产生“镶边效果” (flanging)。实际上这种效果的算法原理与式 4.6 所示的合声延迟效果相同。尽管两者的核心算法是一样的, 但人耳听觉对镶边效果与合声效果的听感是不一样的。对于镶边效果, 延迟时间一般在 0 ~ 10 毫秒之间, 而合声效果则长至 10 ~ 25 毫秒。显然需要重视的是, 所有这里提到的延迟效果都会发生相长和相消干涉 (详见第 4 章第 4 节), 而其中一个有趣的现象称为“梳状滤波” (comb-filtering), 将在第 7 章讨论滤波器和混响器时介绍。

5. 乐例分析

首次使用粒子合成来创作音乐的重要作曲家之一是希腊建筑师兼作曲家伊阿尼斯·克赛纳基斯 (Iannis Xenakis)。克赛纳基斯于 1958 年利用于粒子合成的概念, 用麦克风录制的声音来合成了类似煤燃烧的“噼啪”声。这首作品称为《具像的 PH 值》(Concrète PH), 是 1958 年在布鲁塞尔世界博览会上, 由飞利浦公司展馆所展示的瑞士建筑师勒·柯布西耶 (Le Corbusier) 的多媒体音乐会《电子之诗》(Poème Electronique) 的一部分。但是在观众进出飞利浦公司展馆时放映的以埃德加德瓦·瓦雷泽 (Edgard Varèse) 的《电子之诗》片段为主的多媒体演出中, 这首曲子仅作为节目间的间奏持续了约三分半钟。之所以这是一首非常特别的作品, 不仅因为在上世纪五十年代末, 利用声粒子的概念将成百上千段胶片、声音磁带片段等拼接在一起的技术难度非常大, 而且更重要的、也最为令人惊讶的是, 克赛纳基斯声音粒子合成思想的思考和呈现, 即利用不同的音色、声粒子密度、声音片段和声音素材本身就可以构建一个音乐作品。28 年

后,以粒子合成著称的作曲家巴里·特鲁阿克斯(Barry Truax)创作了一首名为《奔腾的河流》(Riverrun,1986)的作中品——曲名取自于詹姆斯·乔伊斯(James Joyce)的小说《芬尼根的苏醒》(Finnigan's Wake)中的第一个词。该作品表现了河流的奔流与停滞,汹涌与平静,或者水流处于两者之间的微妙状态。从技术层面看,特鲁阿克斯所使用的粒子密度在每秒100~2000粒之间。该作品成功的关键在于作曲家以形成溪流的小液滴作为基本声粒子进行创作,由微至大,直至乐曲最后声音发展为在田野上汹涌奔腾的河流。

本章所讨论的失真、压缩等其他效果在流行音乐录制中相当常见。实际上,压缩是整个现代录音文化和音像制品进入市场前,母带处理阶段的重要特征。合理地利用失真和压缩效果也会给音乐作品带来极具艺术性和创造性的效果。一个以运用这些效果获得特殊音响而闻名的作曲家是美国的特伦特·雷泽诺(Trent Reznor),他的专辑《牙齿》(With Teeth)利用了大量来自乐器和人声的粗糙化(harshness)、失真化和低质化(low-fi)的声音来制造特殊效果。

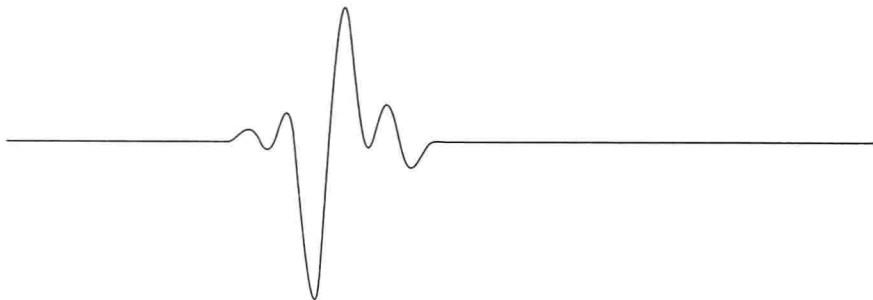
参考文献与扩展阅读

[1] Goben, H. F. 1984. Quantifying Music. The Science of Music at the First Stage of the Scientific Revolution, 1580–1650. Springer.

[2] Gabor, D. 1946. “Theory of Communication”, The Journal of the Institution of Electrical Engineers, 93(3), 429–457.

[3] Gabor, D. 1947. “Acoustical quanta and the theory of hearing”, Nature, 159(4044), 591–594.

第 4 章 正 弦 波



1. 引 言

本书一开始简要地介绍了正弦波,希望从这些我们或多或少熟悉的知识开始进行讨论。本章将运用复数系中功能强大的“欧拉公式”(Euler formula)重新对正弦波的有关概念进行讨论。稍后还将介绍如何使用该公式来处理三角函数和谐波生成器,特别是当我们知道了傅立叶变换之后,我们将在第 8 章对其进行深入探讨。本章的结尾处还将学习一个简单、有效而经济的声音合成算法,即调频合成(frequency modulation synthesis)。最后,还会介绍一些音乐实例和使用本章所述的概念进行创作的作曲家。

2. 正弦曲线

第 1 章由正弦波开始,简要讨论了如式 2.1 所描述的三个特征参数——振幅、频率和初相位。

$$y(t) = A \cdot \sin(2 \cdot \pi \cdot f \cdot t + \phi) \quad (2.1)$$

通常将式 2.1 中时间变量前的部分定义为 ω ——即角频率,单位为弧度/秒(rad/sec),同时频率 f 的单位为赫兹(Hz)。

$$\omega = 2 \cdot \pi \cdot f \quad (2.2)$$

式 2.1 中的相位 ϕ 用于描述波形相对于参考点的相移,当参考点 $t=0$ 时,它更确切地是指“初相位”(initial phase)。上式中的 ϕ 是一个常数。与初相位不同的是,随时间而变化的相位使用“瞬时相位”(instantaneous phase)一词,表达式如式 2.3[注意 ϕ 本身可以是随时间变化的函数 $\phi(t)$]。

$$\Theta(t) = 2 \cdot \pi \cdot f \cdot t + \phi = \omega \cdot t + \phi \quad (2.3)$$

当我们将频率分量和初相位分量合称为瞬时相位时,可能看起来有点混乱,但如果将整合了 ϕ 的频率 f 看成是每个时刻 t 的特定的弧度值(即瞬时相位),上述方程和定义就会成立,即

$\Theta(t)$ 的输出单位弧度明显是 f 与 t 的单位相互抵消的结果。也就是说,下述正弦波由时刻 $t=0$ 至 $t=m \cdot T=m \cdot 1/f_s$ 时刻的一个周期上的总弧度距离等于瞬时相位。对于频率 f 不变的正弦波,瞬时相位的微分方程,即 $\Theta(t)$ 的导数如式 2.4。

$$\frac{d\Theta(t)}{dt} = 2 \cdot \pi \cdot f \quad (2.4)$$

由积分与微分的互换关系,瞬时相位也可看成是“瞬时频率” $f(\tau)$ 由 $\tau=0$ 至 t 的积分。

$$\Theta(t) = \int_0^t 2 \cdot \pi \cdot f(\tau) d\tau \quad (2.5)$$

与固有不变频率 f 不同,而与瞬时相位相似的是,瞬时频率 $f(\tau)$ 是随时间变化的频率分量。瞬时频率的可用式 2.6 表示。

$$f(t) = \frac{1}{2 \cdot \pi} \frac{d\Theta(t)}{dt} \quad (2.6)$$

数字系统中,瞬时频率一般通过给定时间周期内的相位变化来计算。在本章稍后的内容以及第 9 章的相位声码器一节中,读者将会看到这些定义对于理解频率调制合成的重要性。

3. 虚数,复数与欧拉公式

顾名思义,“虚数”(imaginary number)一词所表示的数字是不真实的——即不能被映射在一维数系中的实数所表示。这里所说的一维,是指实数的值只能在一个类线性缩放的尺度上进行增减。例如,若将 0 作为实数系的中心,可向右平移以得到一个值更大的实数,或者向左平移,即往负数方向移动,也可相应找到一个减小的实数。因此,这种数系的排列是一维的,只需要 x 轴就可以表示。而复数却相反,虽然它的产生一开始被认为会使我们的计算难度大大提升,但它的功能却非常强大而有效。事实上,复数系统往往会用合理而简单的方法来解决一些极端困难的问题。

如式 3.1 所示的二阶多项式的两个可能的根如式 3.2:

$$a \cdot x^2 + b \cdot x + c = 0 \quad (3.1)$$

$$x = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \quad (3.2)$$

对于这一二阶多项式,该式求解是完美的,而除了根号部分的值 $b^2 - 4ac < 0$ 时。如,令 $a = b = c = 1$,则得到一个很奇怪的解,如式 3.3。

$$x = \frac{-1 \pm \sqrt{-3}}{2} \quad (3.3)$$

我们基本撞进了一个死胡同,因为用实数系无法定义负数的平方根——而用复数系就能挽救这一局面。若定义如式 3.4 的虚数,就可以处理如负数的平方根这样仅用实数系不能解决的异常问题,即将 -1 的平方根定义为 i 。

$$i = \sqrt{-1} \quad (3.4)$$

如果运用虚数系,上述二项式方程的解如下:

$$x = \frac{-1 \pm \sqrt{1-4}}{2} = -\frac{1}{2} \pm i \frac{\sqrt{-3}}{2} = -0.5 \pm i \frac{3}{2} = -0.5 \pm i1.5 \quad (3.5)$$

因此,复数系中复数的一般形式表示为 $a+ib$, 其中 a 为实部, b 为虚部。也常用字母 j 来代替字母 i , 将复数定义成 $a+jb$ 的形式, 特别是在工程书籍中能避免与常用于电气工程中表示电流的符号 i 相混淆。为了以示区别, 本书将使用 j 作为虚数符号。

如果认为实数是如图 3.1(为了方便说明, 这里以整数为例) 的一维系统, 则复数就是一个如图 3.2 所示的二维数系统。由于复数可用二维平面来表示, 因此可以用三角函数, 即从直角坐标转换为极坐标来表示复数, 如图 3.3。

直角坐标的复数形式为上述 $c=a+jb$ 的形式。同一点 c 的极坐标复数形式可用角度和欧氏距离来表示。

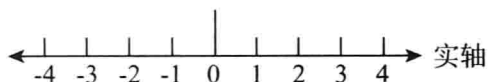


图 3.1 一维系统: 实数

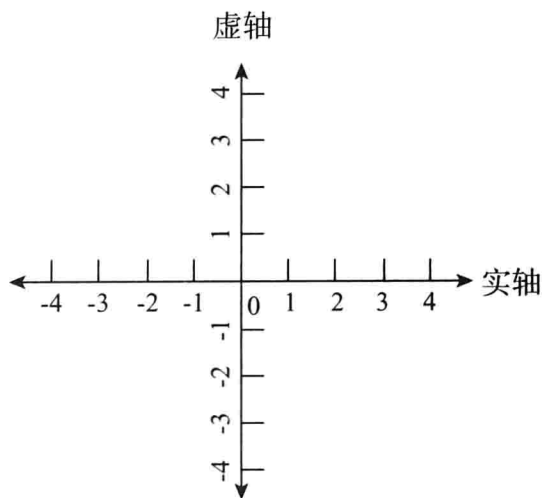


图 3.2 二维系统: 复数

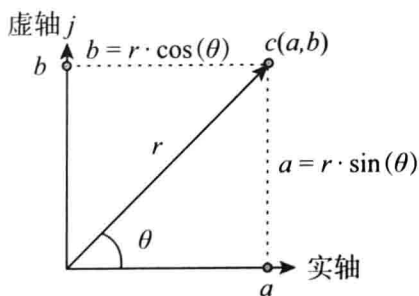


图 3.3 复数的直角坐标与极坐标形式

$$b = r \cdot \cos(\theta) \quad (3.6)$$

$$a = r \cdot \sin(\theta) \quad (3.7)$$

$$r = \sqrt{a^2 + b^2} \quad (3.8)$$

$$\theta = \arctan\left(\frac{b}{a}\right) \quad (3.9)$$

欧拉公式

因瑞士数学家莱昂哈德·欧拉(Leonhard Euler)而得名的欧拉公式,被众多数学家和非数学家们认为是最为优雅和美丽的数学定理之一。达成这一共识的原因之一是该公式能够描述简谐运动、整数、复数及实数。这是一个功能强大的工具,它将有助于处理简谐运动中的困难问题,稍后在本章及其他章节中将对此进行讨论。欧拉公式的定义式如下:

$$e^{j\theta} = \cos\theta + j\sin\theta \quad (3.10)$$

令幅值 $r=1$, 利用欧拉公式可将 $c=a+jb$ 改写成更紧凑的式 3.11。

$$c = r \cdot e^{j\theta} \quad (3.11)$$

利用正弦与余弦函数可将上述指数表达式写成如式 3.12 和式 3.13 所示形式。

$$\cos\theta = \frac{e^{j\theta} + e^{-j\theta}}{2} \quad (3.12)$$

$$\sin\theta = \frac{e^{j\theta} - e^{-j\theta}}{2j} \quad (3.13)$$

将指数分量 $e^{+j\theta}$ 和 $e^{-j\theta}$ 代入欧拉公式,上述式子得证。然而读者现在可能仍不清楚为什么欧拉公式在信号处理中如此重要,但如果继续读下去,我们很快会在下一节看到它是很有用的。

4. 正弦波调制技术(1):调幅

第 4 节与第 5 节将讨论一些经典的正弦波调制技术。我们将看到如何仅通过替换正弦波的振幅和频率分量来获得富有音乐性的素材。我们首先关注正弦波的振幅分量。

4.1 拍频

第 1 章初步讨论了两个或两个以上的波形发生相消或相长干涉的现象,如余弦波的增强、减弱以及在极端情况下完全相互抵消的情况。本节将进一步研究“拍频”(beating)的概念。两列振幅相同而频率不同的余弦波叠加时有如下关系:

$$A \cdot \cos f_1 + A \cdot \cos f_2 = 2 \cdot A \cos\left(\frac{f_1 - f_2}{2}\right) \cdot \cos\left(\frac{f_1 + f_2}{2}\right) \quad (4.1)$$

由式 4.1 可知,两列余弦波叠加等于两者频率和与频率差的均值余弦的乘积。实际上,式 4.1 称为“环形调制”(ring modulation),本章 4.2 节会做简略介绍。若减小两者的频率差 $f_1 - f_2$ 至 1~10Hz 或更小的范围,则在听感上会产生“拍频”。图 4.1 给出了两个采样率为 8kHz,频率分别为 400Hz 和 402Hz 的余弦波叠加后的结果。

此例我们会听到一个被 2Hz ($|f_1 - f_2|$) 的低频进行调制后的平均频率为 401Hz ($(f_1 + f_2) / 2$)

的声音。有趣的是,尽管式 4.1 所示实际应该听到的拍频频率为 $(f_1 - f_2) / 2 = 1\text{Hz}$,但这并不是我们真正听到的频率。原因在于人耳的听觉系统和感知。对于如 1Hz 的低频,实际听到的拍频频率为调制频率的两倍,即:

$$f_{\text{beating}} = 2 \cdot \frac{f_1 - f_2}{2} = f_1 - f_2 \tag{4.2}$$

之所以感觉的是调制频率的倍频,是因为在低频余弦调制器($f_1 - f_2$ 分量)中,过零点数是一个完整周期的过零点数的两倍,如图 4.2。对于每个过零点,由式 4.1 可知叠加后信号的幅值为 0——因此在一个周期中出现了倍频并产生了拍频的感觉。

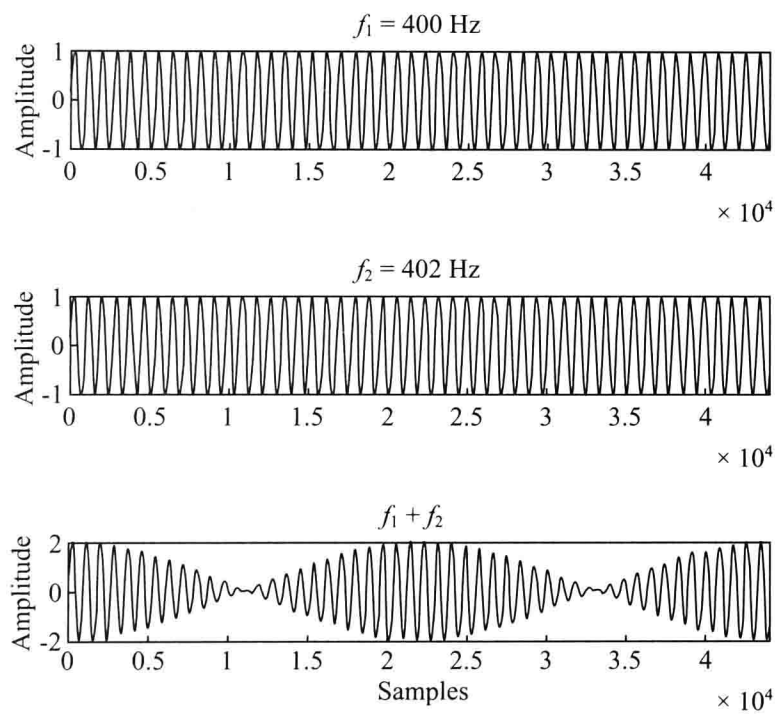


图 4.1 拍频: $f_1 = 400\text{Hz}$, $f_2 = 402\text{Hz}$

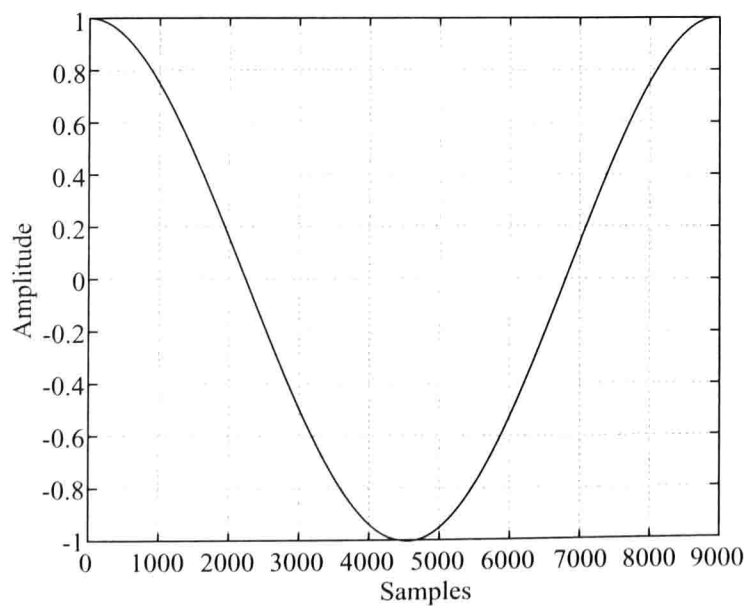


图 4.2 一个余弦周期

至此我们忘了对式 4.1 进行证明。用三角公式来证明可能会有点繁琐,而如果使用欧拉公式,证明式 4.1 的右边等于左边的过程将变得非常直观。首先可将式 4.1 的右边用 $e^{j\theta}$ 因子改写,其中 θ_1 与 θ_2 的定义如下:

$$\theta_1 = \frac{f_1 - f_2}{2} \quad (4.3)$$

$$\theta_2 = \frac{f_1 + f_2}{2} \quad (4.4)$$

将 θ_1 与 θ_2 代入式 4.1,则对于余弦函数,运用欧拉公式有:

$$\begin{aligned} 2 \cdot A \cdot \cos\left(\frac{f_1 - f_2}{2}\right) \cdot \cos\left(\frac{f_1 + f_2}{2}\right) &= 2 \cdot A \cdot \cos\theta_1 \cdot \cos\theta_2 \\ &= 2 \cdot A \cdot \left[\left(\frac{e^{j\theta_1} + e^{-j\theta_1}}{2} \right) \cdot \left(\frac{e^{j\theta_2} + e^{-j\theta_2}}{2} \right) \right] \\ &= \frac{A}{2} \cdot [(e^{j\theta_1} + e^{-j\theta_1}) \cdot (e^{j\theta_2} + e^{-j\theta_2})] \\ &= \frac{A}{2} \cdot [e^{j(\theta_1 + \theta_2)} + e^{j(\theta_1 - \theta_2)} + e^{j(\theta_2 - \theta_1)} + e^{-j(\theta_1 + \theta_2)}] \end{aligned} \quad (4.5)$$

现在用 f_1 与 f_2 代换式 4.5 中的 θ_1 与 θ_2 ,则:

$$\begin{aligned} &\frac{A}{2} [e^{j(\theta_1 + \theta_2)} + e^{j(\theta_1 - \theta_2)} + e^{j(\theta_2 - \theta_1)} + e^{-j(\theta_1 + \theta_2)}] \\ &= \frac{A}{2} \cdot [e^{jf_1} + e^{-jf_2} + e^{jf_2} + e^{-jf_1}] \\ &= \frac{A}{2} \cdot [(e^{jf_1} + e^{-jf_1}) + (e^{jf_2} + e^{-jf_2})] \\ &= A \left[\frac{(e^{jf_1} + e^{-jf_1})}{2} + \frac{(e^{jf_2} + e^{-jf_2})}{2} \right] \\ &= A \cdot \cos f_1 + A \cdot \cos f_2 \end{aligned} \quad (4.6)$$

注意在倒数第二步,我们利用了式 3.12 的余弦恒等式,结果如所预计的那样得到了式 4.1 中两余弦的和。

4.2 振幅调制与环形调制

对于音频,振幅调制 AM (Amplitude modulation) 和环形调制 (ring modulation) 合成技术是两个经典的声音合成方法。正如我们所看到的,振幅调制与环形调制互相仅有微小的差别。此外,这两种与拍频有关的简单算法在音乐创作、声音设计 (sound design) 和音效制作软件中均有能力生成令人感兴趣的音色。

4.3 振幅调制 (AM)

调制一词往往是指对振荡器的频率、相位或振幅等参数进行调整。振幅调制则是对有关振

荡器的振幅进行调整——最简单的设置是用两个振荡器,称为“载波器”(carrier)和“调制器”(modulator)。载波器是被调整的振荡器,而调制器则是指修改载波器的振荡器。计算机音乐(即人耳听觉频率范围)所使用的调幅合成(AM synthesis)的经典定义如式 4.7,其中 f_c 和 A_c 分别是载波器频率及其振幅, f_m 和 A_m 则是调制器的频率和其振幅。

$$y = \cos(f_c) \cdot \{A_m \cdot \cos(f_m) + A_c\} \quad (4.7)$$

$$\begin{aligned} y &= \cos(f_c) \cdot [A_m \cos(f_m) + A_c] \\ &= A_m \cdot \cos(f_c) \cdot \cos(f_m) + A_c \cdot \cos(f_c) \end{aligned} \quad (4.8)$$

式 4.8 是式 4.7 的展开式,它说明调制后信号的振幅可以分为两个部分——经调制部分的振幅和原振幅。也就是说,原载波器通过参数 A_c 整合到了调制后的振幅中。仔细观察式 4.8,根据三角函数的和差定理,式 4.8 还可以改写成式 4.9,即式 4.6 所证明的拍频分量。

$$\begin{aligned} \cos(f_c) \cdot [A_m \cos(f_m) + A_c] &= A_m \cdot \cos(f_c) \cdot \cos(f_m) + A_c \cdot \cos(f_c) \\ &= \frac{A_m}{2} \cos(f_c + f_m) + \frac{A_m}{2} \cos(f_c - f_m) + A_c \cdot \cos(f_c) \end{aligned} \quad (4.9)$$

由上式结果可知,等式右边包含了原载波分量和两个调整了频率及振幅特征的新分量。这就是经典的调幅(AM)合成算法,用于包括早期电影工业的科幻电影中制造机器人声音在内的众多音频应用场合。式 4.7 的离散形式如式 4.10, n 是采样点序号, f_s 为采样率。

$$y[n] = \cos\left(2 \cdot \pi \cdot f_c \cdot \frac{n}{f_s}\right) \cdot \left[A_m \cos\left(2 \cdot \pi \cdot f_m \cdot \frac{n}{f_s}\right) + A_c\right] \quad (4.10)$$

图 4.3 给出了振幅调制中的和差结果,图 4.4 的上图为载波器的波形,中图为调制器,调制后的信号为下图。调制频率低于 10Hz 或更低时,借用音乐上的术语称之为“振音”(tremolo)。因为在这一范围的频率已经远低于人耳听觉频率下限,实际上人耳无法感知出调制频率的音高特征,只能感觉拍频而不是音色上的变化。

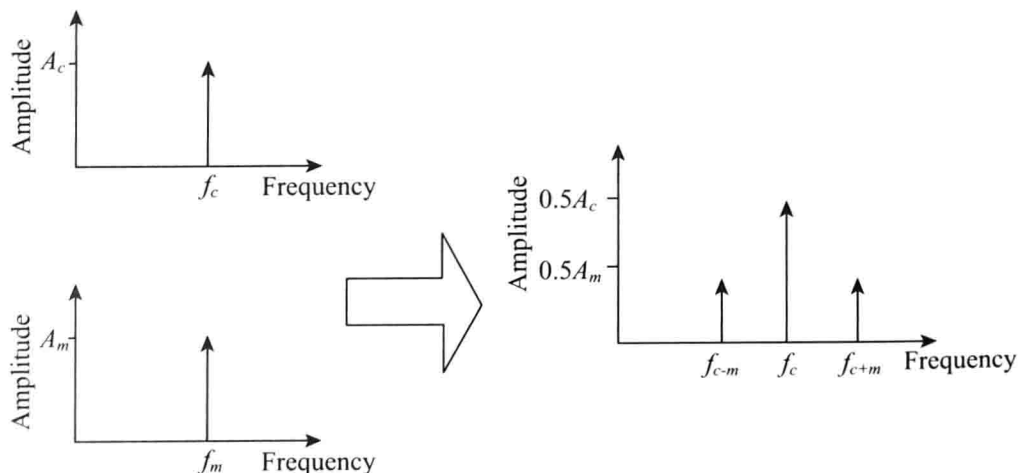


图 4.3 AM 合成输出载波分量的幅值和差

4.4 环形调制

了解了 AM 振幅调制的工作原理,环形调制就可用下式进行定义:

$$y = A_m \cdot \cos(f_m) \cdot \cos(f_c) \tag{4.11}$$

振幅调制和环形调制的不同之处在于,载波器与调制器相乘时,剔除载波分量,输出结果中只保留了和差分量,如式 4.12。

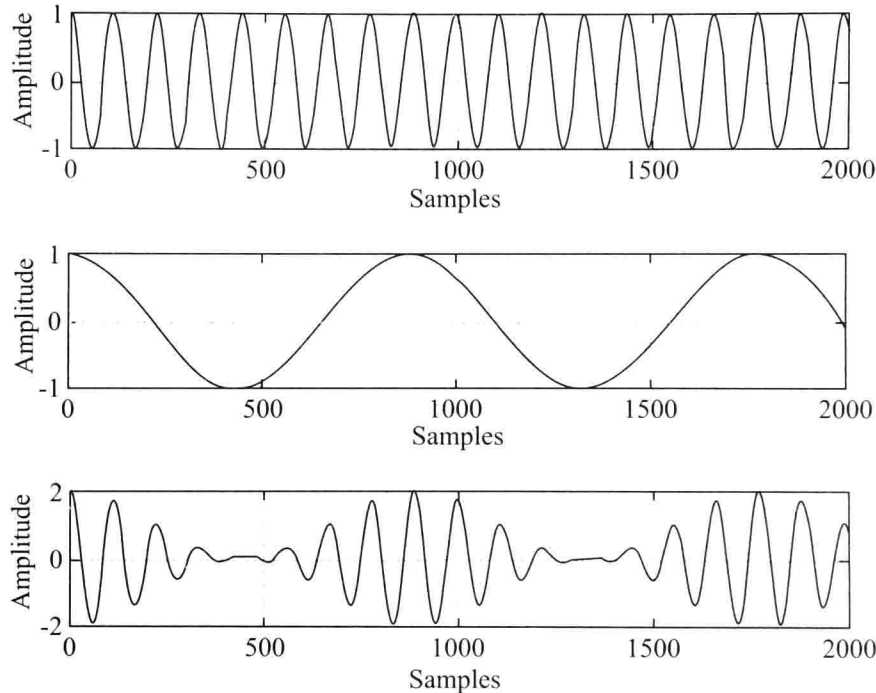


图 4.4 AM 合成, $f_c = 400\text{Hz}$, $f_m = 50\text{Hz}$, $A_c = A_m = 1$, $f_s = 44.1\text{ kHz}$, 载波器(上图), 调制器(中图), 合成输出(下图)

$$y = A_m \cdot \cos(f_c) \cdot \cos(f_m) = A_m \cdot \frac{\cos(f_c + f_m) + \cos(f_c - f_m)}{2} \tag{4.12}$$

式 4.12 的离散形式如式 4.13, n 和 f_s 分别采样点序号和采样率。

$$y[n] = A_m \cos\left(2 \cdot \pi \cdot f_m \cdot \frac{n}{f_s}\right) \cdot \cos\left(2 \cdot \pi \cdot f_c \cdot \frac{n}{f_s}\right) \tag{4.13}$$

如图 4.5 所示,载波分量没有出现在输出信号中,两个振荡器进行调制时,如所预期的只保留了和差分量。

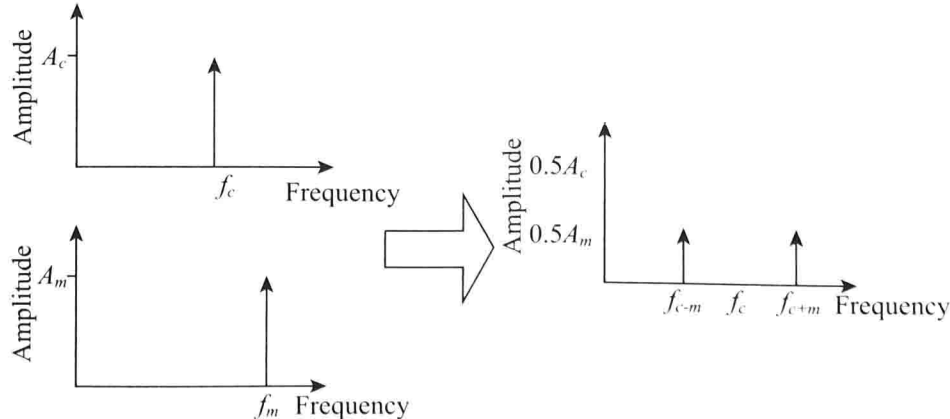


图 4.5 环形调制:输出中载波分量的剔除

4.4.1 复合音的环形调制

环形调制以及振幅调制合成中最令人感兴趣的是,也可以使用基于调制器或载波器的复合振荡器,使输出信号变得更为复杂,并且可能会更有趣。例如,合成后的输出信号可能是一首经管弦乐采样调制后的你最喜欢的歌曲。像这些本身就有复合性的声音信号可被多个经过适当调整的振荡器进行重构。我们将在第8章熟悉傅立叶变换之后,介绍上述声音合成方法的原理。现在,假设组成声音的基础是正弦波,则复合音(特别是周期性的乐音)可被大量特定的正弦分量进行重构。也就是说,如果存在一个足够大的正弦波集合,如2,048个正弦波的集合,就可以通过设定振幅、频率和相位等值来重构一个复合音的波形。然后大量的波形在特定的时间段组合在一起,往往会产生相长或相消干涉,随后重塑输出信号的波形。这种输出信号将非常接近原复合音的波形。显然,问题的关键是如何计算并获得每个正弦分量的振幅、频率和相位,这些将在第8章讨论。

目前假设任何如音乐这样的复合信号都可分解为正弦分量,则可进一步假设,若将两个复合波形相乘,其结果等于一组波形(相当于多个载波器和调制器)相乘,则由式4.12可知,每个具有自己的振幅、频率和相位特性的余弦或正弦波的因子相乘时,一般声音信号的环形调制可用式4.14表示。

$$y[n] = a_m \cdot x_m[n] \cdot a_c \cdot x_c[n] \quad (4.14)$$

此例有两个一般声音信号,分别为具有各自增益因子的调制器和载波器。由乘法交换律可知,相乘因子的顺序不相干,那么从数学层面看,哪一个信号作为调制器以及哪一个信号做为载波器已经变得不再重要。

5. 正弦波调制技术(2):调频

振幅调制中,两个余弦波相乘产生了载波器与调制器频率的和差分量。与振幅调制AM合成不同,频率调制合成(frequency modulation synthesis),或简称FM合成器(Chowning 1973),可以直接调节频率分量而不是幅值分量。简而言之,FM合成如式5.1的形式所示。

$$y(t) = A_{\text{carrier}} \cdot \sin[2 \cdot \pi \cdot f_{\text{carrier}} \cdot t + A_{\text{mod}} \cdot \sin(2 \cdot \pi \cdot f_{\text{mod}} \cdot t)] \quad (5.1)$$

和振幅调制一样,频率调制也包括载波器的振幅与频率分量(A_{carrier} 和 f_{carrier}),以及调制器的振幅与频率分量(A_{mod} 和 f_{mod})。 A_{mod} 通常称为“调制指数”(modulation index),用以确定调制量的大小,当 $A_{\text{mod}}=0$ 为无调制状态,由式5.1可知随后的输出结果只有载波信号。式5.1其实并不如看起来那么复杂,可将它分为以下两个正弦分量,其中 $g(t)$ 是正弦调制函数。

$$y(t) = A_{\text{carrier}} \cdot \sin[2 \cdot \pi \cdot f_{\text{carrier}} \cdot t + g(t)] \quad (5.2)$$

$$g(t) = A_{\text{mod}} \cdot \sin(2 \cdot \pi \cdot f_{\text{mod}} \cdot t) \quad (5.3)$$

你也许会注意到上式并没有直接调制载波频率,而是加入了一个瞬时相位,实际类似于“相位调制”(phase modulation)。而接下来,我们将简要讨论式5.1之所以被称为频率调制的原因。

从音乐家的角度看,像声乐表演这样的音乐演出场合,频率调制很容易被察觉。当歌唱家在持续演唱高音旋律的长音时,往往会用低频振荡来保持音高,我们就会在声乐表演中听到

“颤音”(vibrato)。这实质上就是频率调制。对于传统意义上的频率调制合成,调制频率位于人耳音高感知频率范围内。在这种情况下,不会产生经低频调制的振音感觉,而是在音色上有所变化。这一点与振幅调制相似——调制频率低于人耳音高感知频率范围时,该低频分量在输出信号中就会变成时域事件,人耳能感觉到的是拍频或“振音”(tremolo)。而当调制频率在人耳音高感知频率范围内时,声音的音色就会发生改变。

5.1 调频:旁频与贝塞尔函数

由以上论述可知,FM 合成的独特之处在于所获得的音色。特别是在 20 世纪 80 年代,随着沿用至今的具有传奇色彩的雅马哈(Yamaha)DX 7 数字合成器的出现,这一算法在商业上已经获得了巨大的成功。FM 合成中的音色是通过所谓的“旁频”(sidebands)来控制的。如振幅调制合成能够通过频率的和差分量来描述所输出的合成信号,则这两个分量也可以称为旁频——即以载波频率为中心频率,镶边频率分量 f_1+f_2 和 f_1-f_2 可以看成是载波频率的旁频。在 FM 合成中,大量的旁频分量会使合成后的音色更复杂也更有趣。旁频与载波频率及调制频率有如下关系:

$$f_{\text{sidebands}} = f_{\text{carrier}} \pm f_{\text{mod}} \cdot k \tag{5.4}$$

由图 5.1 可知,这些镶边调制频率分量(即旁频)分布在载波频率 f_{carrier} 两侧。将 FM 合成算法改写成更传统的带有调制指数 β 的表达形式,则有:

$$y(t) = A_{\text{carrier}} \cdot \cos [\omega_{\text{carrier}} t + \beta \cdot \sin(\omega_{\text{mod}} t)] \tag{5.5}$$

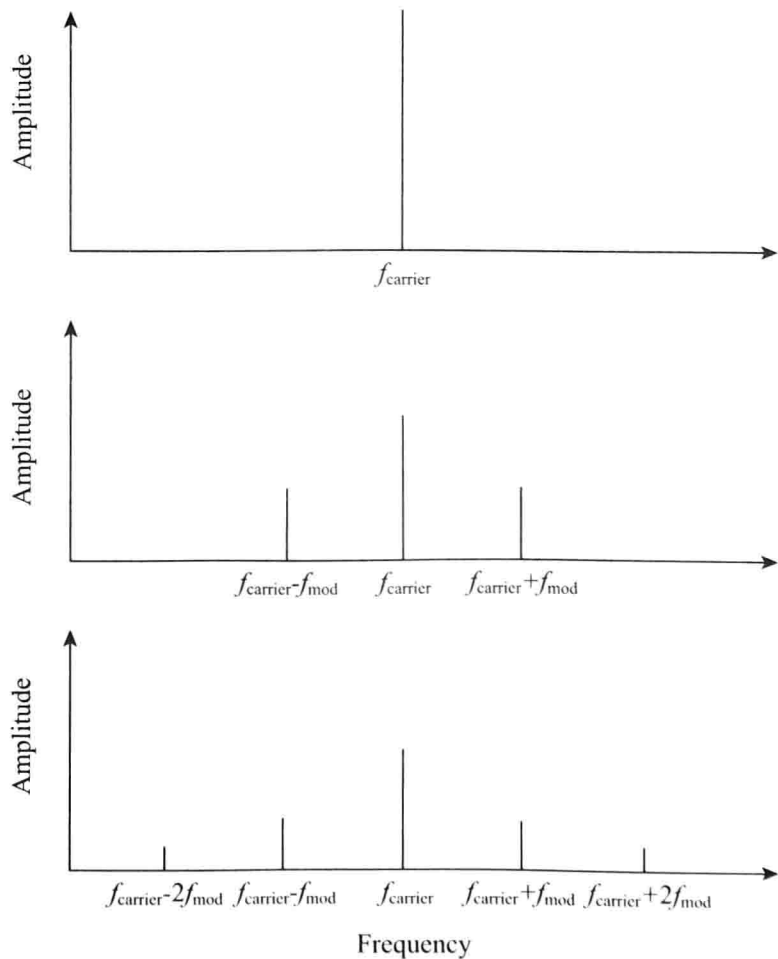


图 5.1 载波(上图),载波和两个旁频(中图),具有多个旁频的载波(下图)

式 5.5 可以进一步用旁频的幅值分量表示,如式 5.6。

$$y(t) = \sum_{p=-\infty}^{p=+\infty} J_p(\beta) \cdot \cos [(\omega_c + p \cdot \omega_m)] t$$

(5.6)

$J_p(\beta)$ 是一个贝塞尔函数 (Bessel function), 它是载波器和旁频分量 $[\cos(\omega_c + p\omega_m)]$ 幅值的权函数。旁频 $\omega_m = 2\pi f_m$ (rad) 由调制频率 f_m (Hz) 控制, 而整数 p 则决定了以载波频率 f_c 为中心分布的旁频的数量。贝塞尔函数的表达式如式 5.7, 贝塞尔函数所使用的参数总结如表 5.1。

$$J_p(\beta) = \sum_{k=0}^{\infty} \frac{(-1)^k \cdot \left(\frac{\beta}{2}\right)^{p+2k}}{k! (p+k)!}$$

(5.7)

式 5.7 往往又可表示为:

$$J_p(\beta) = \sum_{k=0}^{\infty} \frac{(-1)^k \cdot \left(\frac{\beta}{2}\right)^{p+2k}}{k! \Gamma(p+k+1)}$$

(5.8)

其中

$$\Gamma(p) = (p-1)!$$

(5.9)

表 5.1 FM 合成参数

参数	描述
p	旁频序数, $p=0$ 为载波频率, $p=1$ 为 1 次旁频, 此次类推。
k	从 0 ~ K 的整数
$J_p(\beta)$	旁频与载波频率的贝塞尔函数幅值
β	调制指数
注: $0^0 = 1$ $0^q = 0, q > 0$ 且 $q \neq 0$ $0! = 1$	

为了更好地理解贝塞尔函数的作用和工作原理, 令式 5.6 中 $p=0$, 即合成后的输出信号中只包含载波而没有任何旁频, 则有:

$$\begin{aligned} y(t) |_{p=0} &= J_p(\beta) \cdot \cos [(\omega_c + p \cdot \omega_m)] t \\ &= \left[\sum_{k=0}^{\infty} \frac{(-1)^k \cdot \left(\frac{\beta}{2}\right)^{0+2k}}{k! (0+k)!} \right] \cdot \cos [(\omega_c + 0 \cdot \omega_m)] t \\ &= \left[\sum_{k=0}^{\infty} \frac{(-1)^k \cdot \left(\frac{\beta}{2}\right)^{2k}}{k! k!} \right] \cdot \cos(\omega_c) t \end{aligned}$$

(5.10)

上式表明 $p=0$ 时, 只有载波 $\cos(\omega_c t)$, 而没有旁频。 $p=1$ 时, 则在载波频率 f_c 的两侧各加入

了一个旁频: $\omega_c \pm \omega_m$ 。则这些次旁频的幅值也可如上所述通过增加旁频阶数 p 来计算。

$$\begin{aligned}
 y(t) \big|_{p=1} &= \sum_{p=-1}^{p=1} J_p(\beta) \cos[(\omega_c + p \cdot \omega_m)] t \\
 &= J_{-1}(\beta) \cdot \cos[(\omega_c - 1 \cdot \omega_m)] t \\
 &\quad + J_0(\beta) \cdot \cos[(\omega_c + 0 \cdot \omega_m)] t + J_1(\beta) \cos[(\omega_c + 1 \cdot \omega_m)] t
 \end{aligned} \quad (5.11)$$

展开式 5.11 中 $p=+1$ 的旁频分量如下:

$$\begin{aligned}
 &J_1(\beta) \cdot \cos[(\omega_c + 1 \cdot \omega_m)] t \\
 &= \left[\sum_{k=0}^{\infty} \frac{(-1)^k \cdot \left(\frac{\beta}{2}\right)^{1+2k}}{k! (1+k)!} \right] \cdot \cos[(\omega_c + 1 \cdot \omega_m)] t
 \end{aligned} \quad (5.12)$$

观察 $J_p(\beta)$ 的分子部分, 当 $\beta=0$ 而 $p > 0, k \neq 0$ 时, 因 $(\beta/2)^{p+2k}=0$ 而恒等于 0, 只有当 $\beta=0$ 且 $k=0$ 时, 因 $0^0=1$ 而非 0 值。即无调制时 ($\beta=0$ 且 $k=0$) 没有旁频, 载波分量的总能量不包含旁频分量。而随着调制指数的增大, 旁频数量就会增加, 声音复杂度的变化如图 5.1 和图 5.2 所示, 结果导致载波能量分散到了旁频分量中。如令式 5.7 中 $p=0$ 时 $\beta=2$, 则有如下结果:

$$\begin{aligned}
 \sum_{k=0}^{\infty} \frac{(-1)^k \cdot \left(\frac{2}{2}\right)^{2k}}{k! k!} &= \sum_{k=0}^{\infty} \frac{(-1)^k \cdot (1)^{2k}}{k! k!} = \sum_{k=0}^{\infty} \frac{(-1)^k}{(k!)^2} = \left(\frac{1}{1}\right) - \left(\frac{1}{1}\right) \\
 &\quad + \left(\frac{1}{4}\right) - \left(\frac{1}{36}\right) + \left(\frac{1}{157}\right) - \dots
 \end{aligned} \quad (5.13)$$

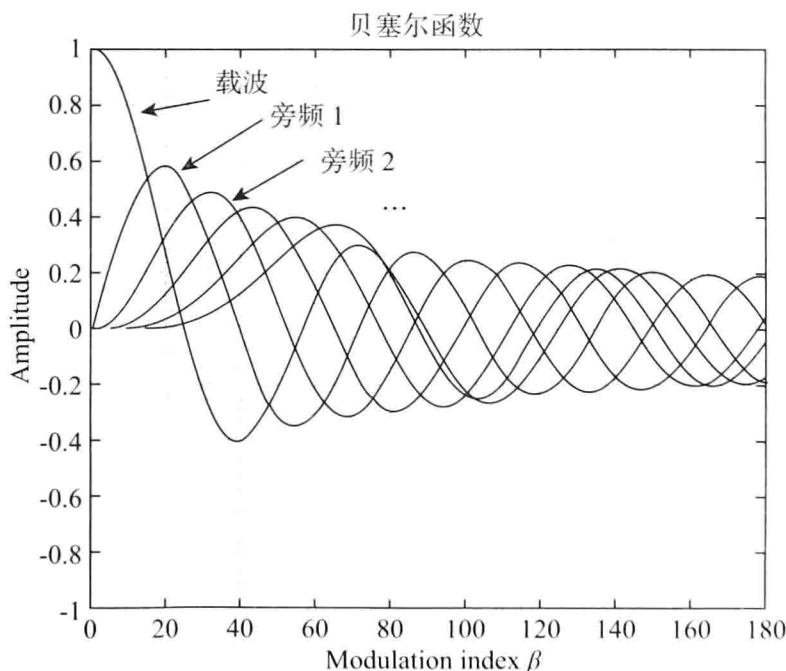


图 5.2 载波和前 5 次旁频的贝塞尔函数

总之, 当 $\beta=0$ 时 (无调制) 只有载波存在, 随着 β 值的增加, 由贝塞尔函数可知, 带宽的增加导致载波中的初始总能量会扩展分散到旁频中。同时, p 值 (旁频阶数) 越大, 贝塞尔函数值越

小。因此,FM 频率调制合成声音的特征是调制指数 β 和调制频率的函数。

由于输出特征是调制指数 β 和调制频率 p 的函数,在乐音合成中,特别是乐器乐音的合成往往会显示出谐波幅值衰减的特点。一般来说,在乐器乐音的谐波结构中,高次谐波(较高的频度分量)能量往往不足,而低频谐波能量较高。如图 5.2 所示,通过贝塞尔函数可模拟 FM 频率调制合成算法的上述特征。

5.2 调制指数

如前所述,FM 合成中调制指数的作用相当于调制器幅值的数量乘法器,它决定旁频的产生并可控制合成声音的音色。根据式 5.14,调制指数 β 通常也可表示为调制器幅值与调制频率的比值。

$$\beta = \frac{A_{\text{mod}}}{f_{\text{mod}}} \quad (5.14)$$

这一调制器幅值与频率的比值可通过求解瞬时相位的积分获得。让我们从如式 5.15 的余弦振荡器开始计算该积分。

$$y(t) = A_c \cdot \cos(2 \cdot \pi \cdot f \cdot t) \quad (5.15)$$

由于 FM 合成中频率受到调制,则频率 f 变成时间函数 $f(t)$ ——即第 5 节开始讨论的瞬时频率 $f(t)$,其由载波频率 f_c 和调制分量 $A_m x(t)$ 表示的定义式如式 5.16:

$$f(t) = f_c + A_m \cdot x(t) \quad (5.16)$$

当调制器 $x(t)$ 本身也是一个余弦波时,调制后的瞬时频率 $f(t)$ 可以写成:

$$f(t) = f_c + A_m \cdot \cos(2 \cdot \pi \cdot f_m \cdot t) \quad (5.17)$$

则信号的瞬时相位可用瞬时频率 $f(t)$ 替换式 5.15 中的频率 f 表示,并对瞬时频率 $f(t)$ 进行积分如下:

$$\begin{aligned} y(t) &= \cos[2 \cdot \pi \cdot f(t) \cdot t] \\ &= \cos\left[2 \cdot \pi \cdot \int_0^t [f_c + A_m \cdot \cos(2 \cdot \pi \cdot f_m \cdot \tau)] d\tau\right] \\ &= \cos\left[2 \cdot \pi \cdot \int_0^t f_c d\tau + 2 \cdot \pi \cdot A_m \cdot \int_0^t [\cos(2 \cdot \pi \cdot f_m \cdot \tau)] d\tau\right] \\ &= \cos\left[2 \cdot \pi \cdot f_c \cdot t + \frac{A_m}{f_m} \cdot \sin(2 \cdot \pi \cdot f_m \cdot t)\right] \end{aligned} \quad (5.18)$$

至此云开雾散,我们发现上式与本小节开始所列出的式 5.1 的形式完全一样。而上式 5.18 的末尾,我们得到了定义为调制器振幅与调制频率之比的调制指数,如式 5.19。

$$y(t) = A_c \cdot \cos\left[2 \cdot \pi \cdot f_c \cdot t + \frac{A_m}{f_m} \cdot \sin(2 \cdot \pi \cdot f_m \cdot t)\right] \quad (5.19)$$

由式 5.19 容易得出其离散形式如式 5.20,其中 n 为采样点序号。这一等式也是软件中常用的 FM 合成算法,

$$y[n] = A_c \cdot \cos \left[2 \cdot \pi \cdot f_c \cdot \frac{n}{f_s} + \frac{A_m}{f_m} \cdot \sin \left(2 \cdot \pi \cdot f_m \cdot \frac{n}{f_s} \right) \right] \quad (5.20)$$

5.3 频率调制的控制参数

不言而喻,调制指数和调制频率是 FM 合成中最重要的参数。一般来说,调制指数越大,旁频扩展越宽。这里以怎样使 FM 合成算法获得的声音听起来像小号声或钟声为例,粗略讨论一些基本的 FM 合成属性。

回顾式 5.16,注意到调制振幅 A_m 控制着以载波频率 f_c 为中心的旁频偏移量,其结果是峰值频率偏移量由调制振幅 A_m 决定,即:

$$\Delta f = A_m \quad (5.21)$$

有趣的是,这意味着以载波频率 f_c 为中心摆动的频率分量并不是调制频率 f_m 的函数,而是调制振幅 A_m 的函数。因此式 5.19 常表示为如下 FM 合成的模拟与数字形式:

$$y(t) = A_c \cdot \cos \left[2 \cdot \pi \cdot f_c \cdot t + \frac{\Delta f}{f_m} \cdot \sin(2 \cdot \pi \cdot f_m \cdot t) \right] \quad (5.22)$$

$$y[n] = A_c \cdot \cos \left[2 \cdot \pi \cdot f_c \cdot \frac{n}{f_s} + \frac{\Delta f}{f_m} \cdot \sin(2 \cdot \pi \cdot f_m \cdot \frac{n}{f_s}) \right] \quad (5.23)$$

这里以峰值频率偏移量 Δf (Hz) 来替换调制振幅 A_m ,使 FM 算法的解释和使用看起来更加直观。

自 1985 年斯克托尔德采用 FM 合成算法来制造有趣的声音或对现有的声学乐器进行调制以来,简单的频率调制算法已在许多方面作了调整。其中一个重要方面是,载波频率与调制频率之比通常称为 $c:m$ 比,其中 c 和 m 分别是载波和调制常数 (Truax 1985)。 $c:m$ 比固然非常重要,因为它会影响旁频特征。因为其他著作对这一参数有很多的讨论,所以本书不会过多论述。对此有兴趣的读者,也可在互联网上搜索有关生成特殊音色的论题。而在结束 FM 合成的讨论之前,我想再强调一下一个控制频率调制 FM 音色质量的重要议题——即如何动态地改变 FM 合成控制参数。以合成铜管的音色为例,重要的是随时间改变参数以获得所控制包络的形态。动态的控制之所以非常关键,是因为在一个声学乐器乐音持续时间内,音质会随时间变化。一般来说,声音的开始阶段(激励部分)拥有丰富的频谱而使声音变得明亮(更高的调制指数),然后过渡到稳态阶段,高频分量和谐波逐渐衰减(声音变得暗淡并需要更少的 FM 旁频)至波形的结束阶段。因此,用包络来控制 FM 算法的参数对于渲染接近声学乐器的声音,以及让声音更加生动和自然是至关重要的。

6. 乐例分析

在音乐中,有无数使用正弦波及其应用的例子,包括拍频和调制的应用。对于不是吉他手的人而言,可能无法听到和看到吉他手们利用两根弦的和声来调弦的情景。原因实际上非常简单,就在于拍频的概念。如果用两根不同的弦同时弹奏同一音高,没有调准时就能听到拍频,反之两弦调准时就不会听到拍频。按住吉他上的一个音品是很简单的,因此使用拍

频原理来代替调音夹进行调弦的原因在于大家只有两只手,很难在保持所弹奏音符的同时调整调音夹!20世纪70年代初,阿尔文·卢西尔(Alvin Lucier)的专辑《双曲线家族沉默中的静止和运动》(*Still and Moving Lines of Silence in Families of Hyperbolas*)就是利用拍频原理进行作曲的例子。在这张专辑的12个作品中,卢西尔让音乐家们以一个或两个固定音高的纯音为参照,演奏十六分音符,音符间用休止符隔开。来自这些音乐家所演奏的音符在音高上轻微提高或降低,以在不同的速度中产生能够听到的拍频效果。总体合成的音符与各音乐家所演奏音符间的频率差越大,则拍频跳动越快——频率差扩大至和谐音程时,拍频消失。这一实例仅仅是阿尔文·卢西尔作品所体现的、从科学领域借用有趣概念进行音乐创作的作曲手法之一。

频率和差分量的概念也被大量应用在音乐演出或创作当中。一个值得注意的人是莱昂·特雷门(Leon Theremin),他是一种被称为“特雷门琴”(Theremin)的单声乐器的发明人。目前发现,许多摇滚乐队——如莱德·齐柏林——都曾使用过这种乐器,但演奏最多的可能是艺术表演家克拉拉·罗克莫尔(Clara Rockmore)。“特雷门琴”是一件与众不同的乐器,它不需要用手接触乐器,只需在乐器的两根天线附近挥手就可以发声。改变可听单音音高的基本原理在于乐器的两个必要因素——和差的概念和电容。“特雷门琴”有两根天线,垂直的一根用于控制音高,水平的一根用于控制振幅。两个振荡器用于调整人耳可听频率范围以外的声音,一个固定而另一个可变。事实证明当手靠近天线时,电容发生轻微改变,振荡器的共振频率动态也随之变化。当手离天线足够远时,振荡频率的动态没有变化,频率和差分量为0Hz,加入了一些高于人耳可听频率范围超高频。总之,手接近天线时,振荡器的共振频率动态产生微小变化,导致频率的和差分量发生改变。两个振荡器的频率叠加后必然会超出人耳可听频率范围,而只有其频率差在人耳可听频率范围内时才会被感觉到。

FM合成被认为是目前最成功的音色处理及合成算法之一,不仅在来自学院或实验机构的学术界,同时在普通大众中,它都受到了广泛认可。该算法的成功不仅是由于它可以合成或模仿现有乐器的声音又或创造“新”的声音,而且还在于它是一个不需要大量昂贵振荡器的经济算法。如果有时光机器回到20世纪80年代,那时的CPU都十分缓慢而且昂贵,就像人们穿着汤普森兄弟牌(Thompson Twins)的正装裤子走路一样。换言之,在FM合成器出现之前,个人计算机用户和广大公众很难使用数字合成器。随着FM合成器的出现,只用两个振荡器和一组包络线就可合成出令人难以置信的丰富声音,整体氛围因此发生了变化。结果出现了大量融合基于FM合成技术声音引擎的PC机声卡,直到后来它或多或少被PCM驱动声卡所取代为止。FM合成技术的倡导者之一是约翰·乔宁(John Chowning),他创作了一些令人难忘的作品,包括《电话》(*Phoné*, 1980~1981),《特伦纳斯》(*Turenas*, 1972)和《纹理》(*Stria*, 1977)等一系列由Recherche协和声学及音乐联合研究所(IRCAM, Institut de Recherche et Coordination Acoustique/Musique)委托,为该研究所题为“20世纪展望”(*Perspectives of the 20th Century*)音乐会所做的作品。其中,《纹理》是一首展现FM合成技术的音色扩展能力的经典作品,同时为声音合成算法在音乐作品创作中的应用提供注解。《纹理》的结构是很严密的,而不是即兴的,它是一个完全用FM合成算法进行组织的作品。该曲的织体和音高结构还利用了黄金分割比。特别是乔宁将载波频率与调制频率的比值 $c:m$ 也设置为黄金分割比。将 $c:m$ 设置为黄金分割比,即将该比值设置为黄金分割比的1至某些指数倍,由此产生的低阶频谱成分展现了黄金分割比的含义。该作品声音效果成功之处,不仅体现在音乐严谨的结构设计水平上,而且通过音乐呈现了

与其光谱信息一致的结构。

在音乐与声音作品中还可找到大师运用环形调制与振幅调制的实例,其中科幻电影中的声音效果就是一个很好的范例,特别是在数字音频技术还未盛行时,那些使用经合成调制处理的音效素材来制造机械声效的例子。在音乐方面,环形调制效果可见于施托克豪斯在1970创作的作品《梵歌》(*Mantra*)。该作品用两台钢琴的声音做为环形调制的载波器,以正弦波为调制信号,以电子声、短波信号声和其他声源做为辅助素材。调制用的正弦波随乐谱变化,制造出了独特而新奇的钢琴声音,其类似于约翰·凯奇(John Cage)作品《准备好的钢琴》(*prepared piano*)的音效。两者的不同在于,凯奇的作品是通过在钢琴的腔体中塞入物理构件来改变单色,而施托克豪斯的《梵歌》中的钢琴声经过了电子调制而不是物理改造。《梵歌》中的环形调制效果是比较细微的,作品通过电子与非电子声音之间的转换来扩展音乐表现力,以引起听众对听觉环境的兴趣。另一个为大众熟知的利用振幅调制来处理电吉他声音的做法,是采用1~7Hz低频正弦波做为调制信号。1990年开播的大卫·林奇(David Lynch)的电视系列片《双峰》(*Twin Peaks*)的开场曲是一个运用振幅调制创作作品的成功范例(特别是其器乐版本)。该作品中,作曲家安吉尔·巴德拉莫蒂(Angelo Badalamenti)利用低频正弦信号来调制一把电吉他的声音[可能是吉他手艾丁·迪克森(Eddie Dixon)或维尼·贝尔(Vinie Bell)演奏],营造出的怪诞情绪十分符合片中所需的气氛。

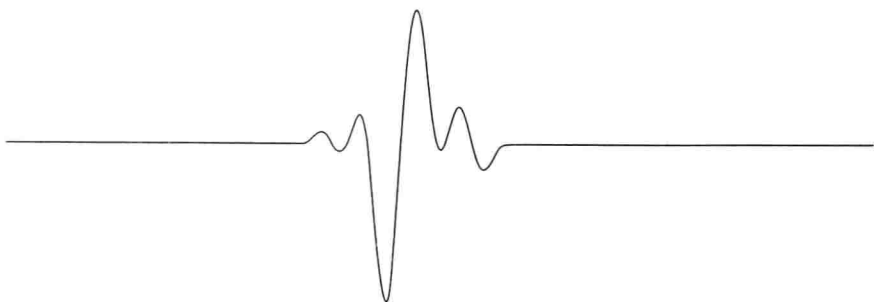
参考文献与扩展阅读

[1] Chowning, J. M. 1973. "The Synthesis of Complex Audio Spectra by Means of Frequency Modulation", *Journal of the Audio Engineering Society*, 21(7), 526-534.

[2] Schottstaedt, B. 1977. "The Simulation of Natural Instrument Tones Using Frequency Modulation with a Complex Modulating Wave", *Computer Music Journal* 1(4), 1977; reprinted in *Foundation of Computer Music*, C. Roads and J. Strawn (eds.). MIT Press, 1985.

[3] Truax, B. 1978. "Organizational Techniques for C: M Ratios in Frequency Modulation", *Computer Music Journal*, 1(4), 39-45; reprinted in *Foundation of Computer Music*, C. Roads and J. Strawn (eds.). MIT Press, 1985.

第 5 章 线性时不变系统



1. 引 言

本章将集中介绍在许多数字信号处理算法中起关键作用的线性时不变系统 LTI (linear time-invariant system)。可以毫不夸张地说,大多数 DSP 技术都是将复杂问题分解成数个子问题,然后对每个子问题进行独立处理后再重组以获得预期效果。符合线性时不变条件的系统,可为上述诸类复杂问题的解决提供简单的方法和手段。描述 LTI 系统最常用的方法是“差分方程” (difference equations)。第 2 章介绍移动平均滤波器的形式时,以及第 3 章介绍合声效果时已经提到了差分方程,因此,在详细定义 LTI 系统之前,本章首先回顾一下第 2 章所介绍的移动平均算法。

2. 差分方程:移动平均算法

第 2 章 5.1 节使用的移动平均算法如式 2.1,也许读者还记得该式用于平滑信号以有利于在计算基频时寻找过零点的位置。

$$y[n] = \frac{x[n] + x[n - 1] + \cdots + x[n - L - 1]}{L} \tag{2.1}$$

$$y[n] = \frac{1}{L} \sum_{k=0}^{L-1} x[n - k] \tag{2.2}$$

以上两式中, n 为采样数, L 是整数常量,用来确定被加总的和按比例伸缩的输入采样点数,它与时窗宽度有关。上述方程实际上可能会比想象的复杂,而很多数学公式往往出于各种原因用较为复杂的形式来表达。仔细观察上述两式可知,它们等价于计算算术平均值的公式,只不过这里以差分方程的形式表达。即差分方程可用来描述某种类型的输入信号与输出信号之间的关系。下面让我们代入一些具体数值来考察差分方程的工作原理。令 $L=2$,输入信号序列 x 如下:

$$x[n] = \begin{cases} n, & 0 \leq n \leq 9 \\ 0, & \text{其他} \end{cases} \tag{2.3}$$

则前几项输出的值 $y[n]$ 如下所示(设 $x[-n]=0$):

$$\begin{aligned}
 & \dots \\
 y[-1] &= \frac{x[-1] + x[-2]}{2} = \frac{0 + 0}{2} = 0 \\
 y[0] &= \frac{x[0] + x[-1]}{2} = \frac{0 + 0}{2} = 0 \\
 y[1] &= \frac{x[1] + x[0]}{2} = \frac{1 + 0}{2} = 0.5 \\
 & \dots \\
 y[9] &= \frac{x[9] + x[8]}{2} = \frac{9 + 8}{2} = 9.5 \\
 y[10] &= \frac{x[10] + x[9]}{2} = \frac{0 + 9}{2} = 4.5 \\
 y[11] &= \frac{x[11] + x[10]}{2} = \frac{0 + 0}{2} = 0 \\
 & \dots
 \end{aligned} \tag{2.4}$$

由上式可知,对于每个输出采样点 n ,算术平均值 $y[n]$ 是以沿输入采样序列分布的两个相邻采样点的值进行计算的。时窗宽度 L 的值越大,则在一个越宽的采样点范围内计算平均值。另外第2章所介绍的跳距的大小,通常也可由整数 L 来确定,而不一定要像上述例子一样设为1。也就是说,如果跳距大小等于时窗口宽度,则时窗按其宽度移动而不是单个采样点。在这一点上需要注意的是,更大的 L 值会产生更小的瞬态响应——即意味着随着时窗宽度的增大会损失更多的时间细节。第2章中提到的瞬态响应指系统对输入信号时间的响应速度。下式给出了用跳距 M 进行控制的移动平均算法。

$$y[n] = \frac{1}{L} \sum_{k=0}^{L-1} x[M \cdot n - k] \tag{2.5}$$

由该式可知,设上述参数 $L=2$ 时,跳距 $M=2$ 而不是1,计算输入样本的移动平均值如下(此处再设 $x[-n]=0$):

$$\begin{aligned}
 & \dots \\
 y[0] &= \frac{x[0 \cdot 2] + x[0 \cdot 2 - 1]}{2} = \frac{x[0] + x[-1]}{2} = \frac{0 + 0}{2} = 0 \\
 y[1] &= \frac{x[1 \cdot 2] + x[1 \cdot 2 - 1]}{2} = \frac{x[2] + x[1]}{2} = \frac{2 + 1}{2} = 1.5 \\
 y[2] &= \frac{x[2 \cdot 2] + x[2 \cdot 2 - 1]}{2} = \frac{x[4] + x[3]}{2} = \frac{4 + 3}{2} = 3.5 \\
 y[3] &= \frac{x[3 \cdot 2] + x[3 \cdot 2 - 1]}{2} = \frac{x[5] + x[4]}{2} = \frac{5 + 4}{2} = 4.5 \\
 y[4] &= \frac{x[4 \cdot 2] + x[4 \cdot 2 - 1]}{2} = \frac{x[8] + x[7]}{2} = \frac{8 + 7}{2} = 7.5
 \end{aligned}$$

$$\begin{aligned} y[5] &= \frac{x[5 \cdot 2] + x[5 \cdot 2 - 1]}{2} = \frac{x[10] + x[9]}{2} = \frac{0 + 9}{2} = 4.5 \\ y[6] &= \frac{x[6 \cdot 2] + x[6 \cdot 2 - 1]}{2} = \frac{x[12] + x[11]}{2} = \frac{0 + 0}{2} = 0 \\ &\dots \end{aligned} \tag{2.6}$$

此例跳距 $M=2$ 时, $y[n]$ 的非零输出值变少了, 即上例跳距 $M=1$ 时, $y[n]$ 的非零输出值为 10 个, 而该例 $M=2$ 时为 5 个。这一结果是可以预知的, 因为 $M=1$ 时逐次计算每个采样点, 系统输出更新速率快, 当 $M=2$ 时, 每隔两个采样点计算一次, 输出更新速率慢。对于采样点少、时窗宽度 L 小的信号, 这未必是太重要的问题, 但对于采样率高而采样点多的信号, 就变成了一个重要的问题。因此不得不慎重选择合适的时窗宽度 L 和跳距 M 。

2.1 因果性

在上述平均移动的例子中, 可以注意到, 当 $n < 0$ 时没有输出, 即输出为 0。这种系统称为因果系统 (causal systems)——当且仅当 $n=0$ 这一“电源开关”接通时, 系统才会输出非零值。也就是说, 如果输出完全取决于当前时刻及过去时刻的输入值时, 则该系统具有因果性。因此, 非因果系统 (non-causal systems) 也可视为有一定的初始状态的系统, 即输入信号进入系统前就有非零和受影响的输出。图 2.1 描述了因果系统与非因果系统的概念: 上图为输入信号序列, 中图为因果系统的输出, 当且仅当 $n=K$ 时刻起有输入时系统才有输出, 下图为非因果系统的输出, 在系统接收输入之前就已经有输出了。

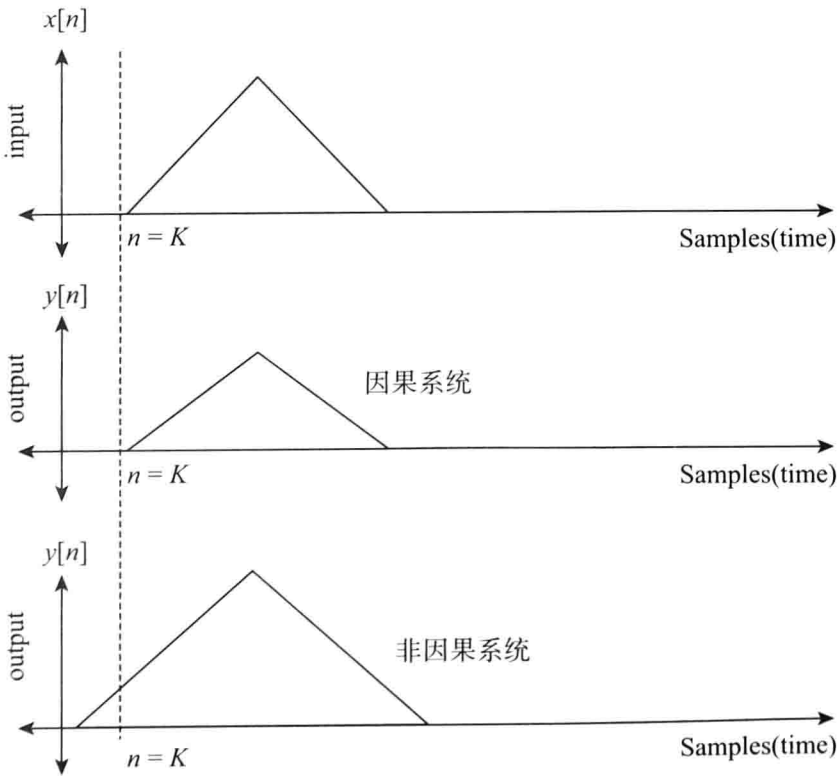


图 2.1 因果系统与非因果系统

2.2 差分方程的一般形式

上例的移动平均差分方程称为非递归差分方程(non-recursive difference equation),因为它没有反馈分量——整数 $m > 1$ 时,不包含写成 $y[n-m]$ 形式的 y 分量,而包含递归部分与非递归部分的差分方程的一般形式如式 2.7:

$$y[n] = a_1 y[n-1] + a_2 y[n-2] + \cdots + a_N y[n-N] + \cdots + b_0 x[n] + b_1 x[n-1] + b_2 x[n-2] + \cdots + b_L x[n-L] \quad (2.7)$$

式 2.7 可用求和形式缩写成式 2.8:

$$y[n] = \sum_{k=1}^K a_k y[n-k] + \sum_{k=0}^L b_k x[n-k] \quad (2.8)$$

$y[\cdot]$ 右边的前一部分具有递归特征,后一部分 $x[\cdot]$ 具有非递归特征。一般差分方程表达了输入、输出信号之间关系,它基于位于时刻 n 的当前和过去的输入样本,以求和与乘积的形式来描述一个复杂系统。

式 2.7 和式 2.8 是差分方程的一般形式,正如我们现在所知,移动平均算法只是差分方程的一个特例,即没有反馈分量且伸缩因子 b_k 等于 $1/L$ 。因此,移动平均类型的差分方程与第 2 章 5.1 节所介绍的信号均值平滑过程一样简单。从本质上讲,通过设定差分方程的伸缩因子(又称为系数或权重)和长度(又称为阶数或宽度),可设计具有特定行为的系统,称为“滤波器”(filter)。这就是采用移动平均算法的滤波器称为移动平均滤波器的原因。滤波器的详细讨论见第 7 章。

需要注意的是,移动平均差分方程是基于输入信号的系数 a_k 和时延 k 而设计的。下节将介绍差分方程为何满足线性时不变性(LTI),并可用于设计如移动平均系统那样可“无故障”工作的系统——而不必担心因输入的时延或乘法处理所带来的非线性输出。

3. 线性时不变 LTI 系统

试想一个由贝斯吉他手、电吉他手和鼓手所组成的三人乐队,在家用录音棚中录音,由于预算较低没有更多的话筒,只能有一只话筒用于贝斯吉他和电吉他,两只用于鼓组。而出人意料的是,用于排练的木制桑拿房式录音间的声音听起来很“平”(flat),并且每个演奏者又必须轮流在一个数字音频工作站(DAW)上录制各自的部分。事不凑巧,贝斯手最后不能如约前来,只能先录制其他人的部分,第二天再录制贝斯手的部分,然后将各部分内容进行合成。伴随着咖啡和只有几个小时的睡眠,在录音棚里工作了 15 个小时之后,三个乐手终于准备通过功率放大和衰减(相乘)对信号(鼓组、贝斯和吉他)进行终混(合成),并最终将各轨混成两个声道(立体声的左声道与右声道)。这样,三个乐手第一次监听到了他们乐队没有同时演奏而是混合而成的声音,乐队鼓手总是会说,“效果比我想象的更糟糕”——其实他所期望的是,尽管演奏家们在不同时间分开演奏各自的乐器而不是一起以乐队形式演奏,混音后听起来的感觉也应该如他们在同一时间同一地点同时演奏一样。在混音过程中,首要的任务是使用大量的均衡器和效果器来渲染一盘听觉印象极其深刻的样带。让两者的听感可能相同的原因是——在不同时间独立录制各个乐器,而在混音时根据线性时不变特征不再添加人为失真——这就是本节所要讨论的议题。

3.1 线性:伸缩不变性与可加性

顾名思义,术语“线性”(linearity)一词来自于“线”(line)一词,它包括基本上呈现出如下关系的任何函数——其输入与输出的关系图像构成一条直线,如图 3.1 所示。

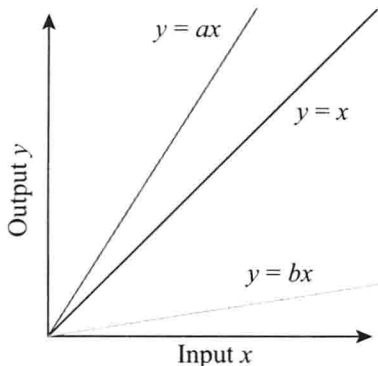


图 3.1 线性

更仔细地考察线性的特征可知,线性函数满足两个基本性质——伸缩不变性[scalability,即同质性(homogeneity property)]和可加性[superposition,即可加和性(additive property)]。线性的伸缩不变性或许是两个性质中较容易理解的一个,其满足的条件如式 3.1 所示。

$$T(\alpha \cdot x[n]) = \alpha \cdot T(x[n]) = \alpha \cdot y[n] \quad (3.1)$$

其中, $T(x[n])$ 是输入 $x[n]$ 的输出, α 是伸缩因子。线性函数的可加性也可用可加和性表示如式 3.2 和式 3.3。如果某个系统同时满足伸缩不变性与可加性条件,则称该系统是线性的。

$$x_3[n] = x_1[n] + x_2[n] \quad (3.2)$$

$$\begin{aligned} T(x_3[n]) &= T(x_1[n] + x_2[n]) = T(x_1[n]) + T(x_2[n]) \\ &= y_1[n] + y_2[n] = y_3[n] \end{aligned} \quad (3.3)$$

总之,上述性质说明无论系统的输入是彼此分离的,还是所有输入预先组合(加和)再一同输入系统的,该系统的输出不变。好比吉他无论是同时录制在一个音频轨上,或是分开在不同音频上录制最后再合成——其结果是一样的。其实,通常的做法是将鼓组中的军鼓、吊钹、中鼓和铜钹都进行分轨录音,以便于录音师在终混和录音棚录音过程的产品制作阶段,能灵活控制录音和混音的效果。

3.2 时不变性:时移不变

时不变系统的特征是输出对时移输入的依赖性。在这样的系统中,输入分量的时移有对应的时移输出。如果系统满足下式条件,则该系统具有时不变性。

$$y_1[n] = x_1[n] = x[n-L] = y[n-L] \quad (3.4)$$

因此,如果时移输入 $x_1[n] = x[n-L]$ 产生的输出 $y_1[n]$ 具有长度为 L 个采样点的时延,即此时输出为 $y[n-L]$,则该系统是时不变的。换言之,输入信号有时延则对应的输出也会有相同的时延。这好比上例中的贝斯手,她晚一天结束的贝斯音轨录制如他们同时一起演奏一样,不会影响最后的混音——即时不变系统不会受信号输入至系统的时间的影响。同时满足线性和时不变性的系统称为 LTI 系统。

让我们通过一些实例来更好地了解 LTI 系统,下面是一个经典例子:令 $y[n] = x^2[n]$,接下来将通过考察其可加性与伸缩不变性来检验它的线性特征。对于伸缩性,证明该系统不可伸缩——将输入加入系统之前乘以伸缩因子 α ,并且系统本身也乘以同样的因子 α ,然后比较两者的结果。

$$T(\alpha \cdot x[n]) = \alpha^2 \cdot x^2[n] = \alpha^2 \cdot y[n] \quad (3.5)$$

$$\alpha \cdot T(x[n]) = \alpha \cdot x^2[n] = \alpha \cdot y[n] \quad (3.6)$$

显然,式 3.5 和式 3.6 的结果不同,这说明该系统已经没有线性特征,而是非线性的。还可以代入其他一些具体数值来验证该系统没有伸缩性。如令 $\alpha=2$ 且 $x[n]=6$,则有 $2 \cdot 6=12$,最后输出为 $12^2=144$ (即伸缩输入并将其加入到该系统中),而具有伸缩性的系统乘以该因子 α 后得到的输出则为 $2 \cdot (6^2)=72$ 。尽管该系统已经不是线性的,但现在仍可检验其可加性。对于可加性,令 $x_3[n] = x_1[n] + x_2[n]$,因 $T(x_1[n] + x_2[n])$ 不等于 $y_1[n] + y_2[n]$ (见式 3.3)可证明该系统的可加性不成立。即式 3.7 不等于式 3.8——两者相差了一个额外的项 $2 \cdot x_1[n] \cdot x_2[n]$ 。既然该系统亦不满足可加性,因此不是线性的。

$$\begin{aligned} T(x_3[n]) &= T(x_1[n] + x_2[n]) = (x_1[n] + x_2[n])^2 \\ &= x_1^2[n] + 2 \cdot x_1[n] \cdot x_2[n] + x_2^2[n] \end{aligned} \quad (3.7)$$

$$y_3[n] = y_1[n] + y_2[n] = x_1^2[n] + x_2^2[n] \quad (3.8)$$

对于时不变性的证明,可通过式 3.4 进行,并比较输入和输出过程。显然,由式 3.9 和 3.10 可知该系统虽然是非线性的,但却是时移不变的,即该系统是非线性时不变系统(non-LTI system)。

$$y_1[n] = x_1[n] = x^2[n-L] \quad (3.9)$$

$$y[n-L] = x^2[n-L] \quad (3.10)$$

另一例如式 3.11。由观察可知该系统是线性的,因此我们将忽略其线性特征的证明。

$$y[n] = x[n/M] \quad (3.11)$$

而接下来对时不变性的证明,如前例处理方法得到的结果,如式 3.12 和式 3.13。由两式可知两个等式的结果不相等,则该系统是时变的。即这两个例子实际上都不是线性时不变系统。

$$y_1[n] = x_1[n] = x[n/M-L] \quad (3.12)$$

$$y[n-L] = x[(n-L)/M] \quad (3.13)$$

使用上述两例证明线性时不变 LTI 的方法可发现本章一开始所介绍的移动平均滤波器是一个 LTI 系统——它实际上是输入的延迟采样点数和系数都等于 $1/L$ 的集合。

3.3 数字信号处理中 LTI 系统的重要性

如果系统是线性的,则其具有伸缩性和可加性的特质。这对于数字信号处理是非常重要的,因为大多数问题都需要按照“先分解后综合”(divide and conquer)的思路来解决。一个复杂系统只有具备线性和时不变性,才可分解为若干更小更简单的子系统。这种思想是强大的,因

为复杂问题往往可以分解并按既定顺序重组,最后再将更简单、更小且更易于管理的子问题重组以获得最终的结果,如图 3.2 所示。



图 3.2 LTI 系统的分解与综合

如果每个子系统都是线性时不变的,则这一思路可以推广到多输入系统 (multi-input system) 中。一个多输入系统的例子如图 3.3 所示,即第 3 节一开始所述的乐队录音的情况。因此,如果每个子系统都是线性时不变的,则任何多输入和多输出的系统都是线性时不变的。

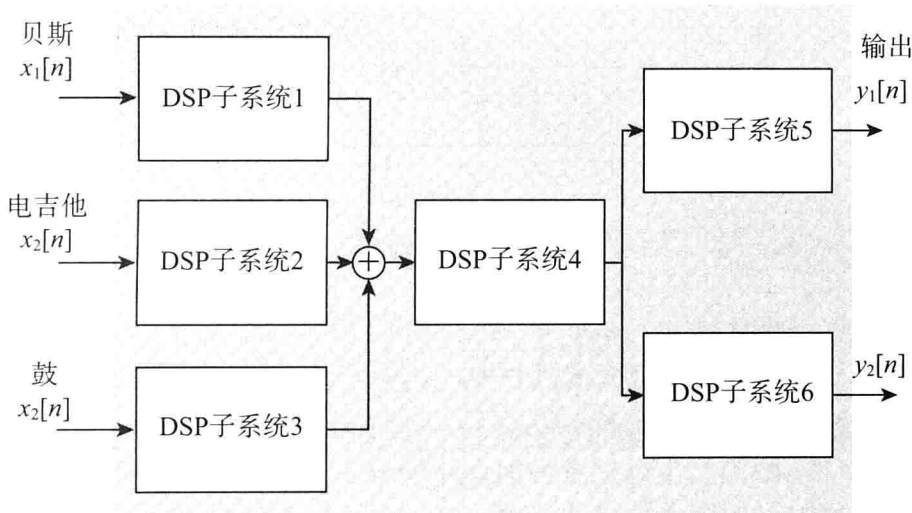


图 3.3 多输入与多输出 LTI 系统

4. 脉冲响应

许多读者或许会记得,自己小时候在花园里发现一只可怕的蜥蜴或一些奇怪动物,当你只是看着它但无法确定其种类时,会怀疑这个东西是不是活的。至少笔者在发现一些不寻常却无法识别的东西时并不会太害怕,而是有时会用棍子去戳它一下,看看它是否是活的,或者只是一根枯树枝,又或者是其他奇形怪状的东西的一部分。这就是一种“脉冲响应”(impulse response),即激励一个系统或上述事件并观察它被激发后的特征——如上述观察蜥蜴状物体的情形,用棍子戳它之后,结果可能是蜥蜴迅速地跑开或在极端情况下咬你一口!

这种用来观察系统输出的“激励”(poking)在信号处理中称为脉冲(impulse)、单位脉冲(unit impulse)或 δ 函数(delta function),它们指的都是同一个事物。从理论上讲,脉冲的时长无限短(时域),幅值无限高(振幅),其时域面积积分等于 1。这种脉冲显然是一个理想概念,在现实世界并不存在。脉冲的定义如式 4.1 所示,用希腊字母 δ 来表示。将输入设为单位脉冲来激励系统可观察系统的脉冲响应。

$$x[n] = \delta[n]$$

(4.1)

δ 函数只在一个采样点上定义,幅值归一化为 1,而其他时间序列上的点的幅值为 0,如式 4.2 和图 4.1。

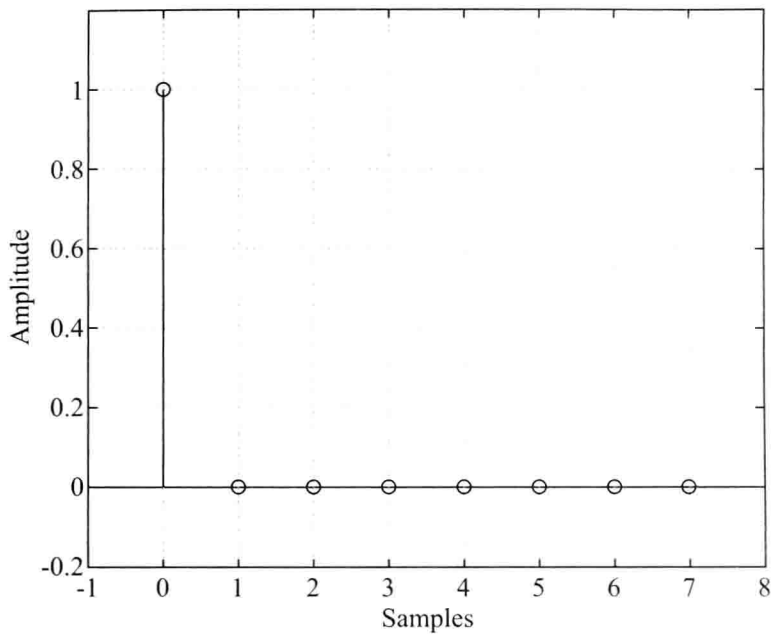


图 4.1 脉冲信号

$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & \text{其他} \end{cases} \quad (4.2)$$

设有未知数字系统,只知道它是非反馈系统。即它没有延迟其差分方程,如式 4.3:

$$y[n] = \sum_{l=0}^L b_l \cdot x[n-l] \quad (4.3)$$

设 $x[n] = \delta[n]$, 将脉冲作为上述系统的输入并观察它的系统输出或脉冲响应,则式 4.3 可写成式 4.4。

$$y[n] = \sum_{l=0}^L b_l \cdot \delta[n-l] \quad (4.4)$$

由 δ 函数的定义,进一步观察系统的输出,则结果如以下形式所示:

$$\begin{aligned} y[0] &= \sum_{l=0}^L b_l \cdot \delta[0-l] \\ &= b_0 \cdot \delta[0] + \cdots + b_L \cdot \delta[0-L] \\ &= 1 \cdot b_0 + 0 + \cdots + 0 = b_0 \end{aligned} \quad (4.5)$$

$$\begin{aligned} y[1] &= \sum_{l=0}^L b_l \cdot \delta[1-l] = b_0 \cdot \delta[-1] + b_1 \cdot \delta[0] + \cdots + b_L \cdot \delta[1-L] \\ &= 0 + 1 \cdot b_1 + \cdots + 0 = b_1 \end{aligned} \quad (4.6)$$

...

...

$$\begin{aligned}
 y[L] &= \sum_{l=0}^L b_l \delta[l-L] = b_0 \cdot \delta[-L] + \cdots + b_{L-1} \cdot \delta[1] + b_L \cdot \delta[0] \\
 &= 0 + \cdots + 0 + 1 \cdot b_L = b_L
 \end{aligned} \quad (4.7)$$

在信号处理中,上述计算所得的脉冲响应记为 $y[n] = h[n]$,当输入为脉冲样本时,则有:

$$x[n] = \delta[n] \quad (4.8)$$

$$y[n] = h[n] \quad (4.9)$$

由式 4.6 和 4.7 可知 $h[n]$ 如下式 4.10。

$$h[n] = \begin{cases} b_n, & n = 0 \cdots L \\ 0, & \text{其他} \end{cases} \quad (4.10)$$

则该系统的输出与输入的关系可写成:

$$x[n] = \delta[n] \rightarrow y[n] = h[n] \quad (4.11)$$

式 4.3 可写成:

$$h[n] = \sum_{l=0}^L b_k \cdot \delta[n-l] = \sum_{l=0}^L h_k \cdot \delta[n-l] \quad (4.12)$$

以上论述也许并没有显示出脉冲响应的重要性,而在第 5 节我们熟悉卷积(convolution)之后,就会看到如果已知系统的脉冲响应($h[n]$),就可判断对任何给定的输入序列的系统输出。脉冲响应的重要性在于不仅能判断系统的输出,还可用来考察系统的其他特征,包括稍后将简要介绍、并将在第 6 章学习频率响应之后会详细叙述的系统稳定性问题。

现实世界中,也可以体验到脉冲响应的存在。例如在你现在所处的任何房间里大声地击掌(只击一次),就可得到该房间的脉冲响应特征(希望你不是在图书馆或拥挤的公共浴室里)。刚才你所做的是用一个脉冲(该例脉冲=击掌声)激发该系统(即该房间),而你所到的就是脉冲响应。需要注意的是在不同类型的空间(或者是同一个房间的不同位置),如有瓷砖地板的浴室,或铺着厚地毯的房间,又或是在你的汽车里,会产生不同的脉冲响应类型。虽然击掌声不是一个上述定义的理想单位脉冲,但该实验能让你很好地理解单位脉冲和脉冲响应在数字信号处理中的作用和重要性。

4.1 有限脉冲响应(FIR)与无限脉冲响应(IIR)

有限和无限脉冲响应系统(finite and infinite impulse system)即 FIR 和 IIR 系统,是指具备有限和无限输出序列的系统。以脉冲激励系统时,若系统存在一个输出恒为 0 的状态,则该系统称为 FIR 系统。由一般差分方程可知:

$$y[n] = \sum_{k=1}^N a_k y[n-k] + \sum_{k=0}^L b_k x[n-k] \quad (4.13)$$

若去除上式的反馈分量 $y[n-k]$,则有:

$$y[n] = \sum_{k=0}^L b_k x[n-k] \quad (4.14)$$

再由脉冲响应的定义可知,一般 FIR 系统的脉冲响应可用式 4.15 表示:

$$h[n] = \sum_{k=0}^L b_k \delta[n-k] \quad (4.15)$$

而系统的输出不恒为 0 或不存在一个输出恒为 0 的状态时,该系统称为 IIR 系统。将 $y[\cdot]$ 用 $h[\cdot]$ 代入, $x[\cdot]$ 用 $\delta[\cdot]$ 代入,由一般差分方程可直接得出该 IIR 系统的脉冲响应如式 4.16 所示:

$$h[n] = \sum_{k=1}^N a_k h[n-k] + \sum_{k=0}^L b_k \delta[n-k] \quad (4.16)$$

为了方便实际运用,在数字系统中,稳定的 IIR 系统最终将到达某个“零态”(zero state),此时幅值很小的输出将被忽略。不过 IIR 系统也可能变得不稳定并“崩溃”(blow up)——如扬声器圆锥筒会因过多的反馈而破裂。至于通常所指的递归与非递归系统特征,或者是反馈与反馈前特征是因 IIR 系统具有反馈分量而产生的, FIR 系统则没有。

4.2 稳定性与 IIR 系统

关注 FIR 和 IIR 系统的另一个重要问题是稳定性。如果系统稳定,则会在某个时刻只输出零值,相当于关闭“电源开关”;如果系统不稳定,则其输出幅值会不断增大,并最终导致系统的破坏和“崩溃”。如前一小节所提到的设置有扬声器和传声器的音乐会就是一个有时能听到的反馈系统——如果传声器的输入信号成为了该扩声系统的输出,而该输出又成为了传声器的输入,则会导致恶性的信号放大反馈。这一结果就是糟糕的“啸叫”声,在极端情况下甚至可以损坏扬声器和调音台等音频设备,甚至更严重的还可能损害人耳。

设一个 IIR 系统的差分方程如下:

$$y[n] = x[n] + a \cdot y[n-1] \quad (4.17)$$

为了计算上述系统的脉冲响应,根据一般情况将输出和输入分别用 $h[\cdot]$ 和 $\delta[\cdot]$ 代入,则有:

$$x[n] = \delta[n] \quad (4.18)$$

$$y[n] = h[n] \quad (4.19)$$

则式 4.17 的差分方程写成该系统的脉冲响应如下:

$$h[n] = \delta[n] + a \cdot h[n-1] \quad (4.20)$$

对于因果系统(只有打开“电源开关”系统才有响应),则有:

$$h[n] = 0, n < 0 \quad (4.21)$$

则脉冲响应序列的输出结果为:

$$h[0] = \delta[0] + a \cdot h[-1] = 1 + 0 = 1$$

$$h[1] = \delta[1] + a \cdot h[0] = 0 + a \cdot 1 = a$$

$$\begin{aligned} h[2] &= \delta[2] + a \cdot h[1] = 0 + a \cdot a = a^2 \\ &\vdots \\ h[k] &= \delta[k] + a \cdot h[k-1] = a^k \end{aligned}$$

(4.22)

由于上述系统是一个 IIR 系统,因此需要检验它的稳定性,否则反馈可能会失去控制,并且 $|a| > 1$ 时该系统不稳定。即 a 绝对值大于 1 时,该系统的输出幅值最终开始无限制地增大。若 $|a| < 1$,则系统稳定并且可以说是有界的。 $|a| > 1$ 时该 IIR 系统不稳定的原因在于式 4.22 中的 a^k 值。即 $|a| > 1$ 时,随着 k 值的增大, $|a^k|$ 的值也增大且无界。

需要注意的是,由于存在上述反馈分量,如果不加注意,IIR 系统存在变得不稳定的可能性。而 FIR 系统因不会产生反馈则总是稳定的——关闭开关时,一个 FIR 系统最终将在某个时刻停止工作。稳定性特征的总结如表 4.1 所示。

表 4.1 FIR 与 IIR 系统对比

	是否有反馈分量?	是否存在恒零值状态?	是否稳定?
FIR	否	是	是
IIR	是	否	否

5. 卷 积

本节将详细研究“卷积”(convolution)这个在数字信号处理和音乐应用软件中非常重要的概念。如第 4 节开始所提到的,如果已知系统的脉冲响应而不知道它的差分方程,可通过卷积定理来求解该系统的输出。

接下来首先讨论卷积的定义,并解释其含义,以及卷积为什么常常出现在数字信号处理(尤其是音频信号处理)中。既然已经了解了 LTI 系统是如何工作的,那么如果一个系统是线性时不变的,输入为单位脉冲时产生的脉冲响应如下所示:

$$\delta[n] \xrightarrow{\text{yields}} h[n]$$

(5.1)

对于 LTI 系统,带有延迟的 δ 函数将产生带有延迟的脉冲响应,如式 5.2:

$$\delta[n-m] \xrightarrow{\text{yields}} h[n-m]$$

(5.2)

同理可知,任何输入序列可用带有权重和延迟的样本表示,则有:

$$x[n] = \sum_{m=-\infty}^{m=+\infty} x[m] \cdot \delta[n-m]$$

(5.3)

当且仅当 $n=m$ 时,式 5.3 的 $x[n]$ 存在,则对于 LTI 系统下式成立。

$$x[m] \cdot \delta[n-m] \xrightarrow{\text{yields}} x[m] \cdot h[n-m]$$

(5.4)

最后由线性中的可加性可知,对式 5.4 的两边求和得式 5.5:

$$\sum_{m=-\infty}^{m=+\infty} x[m] \cdot \delta[n-m] \xrightarrow{\text{yields}} \sum_{m=-\infty}^{m=+\infty} x[m] \cdot h[n-m]$$

(5.5)

上述关系就称为卷积。如果令式 5.5 的右边等于 $y[n]$, 则卷积的一般定义式如式 5.6, 其中星号($*$)为卷积运算符号, 称输入 x 以 h 进行卷积, 其中 h 是系统的脉冲响应。因此, 如前所述, 如果已知系统的脉冲响应而其差分方程未知, 可仅以其脉冲响应和任何输入序列来计算该系统的输出。

$$y[n] = x[n] * h[n] = \sum_{m=-\infty}^{m=+\infty} x[m] \cdot h[n-m] \quad (5.6)$$

卷积的一个重要应用通常称为“卷积混响器”(convolution-based reverb)。如果读者实施了上述在房间中击掌的实验, 就能意识到房间的脉冲响应可录制并存储为一个数字序列 $h[n]$ 。如果在一个大教堂中击掌, 就会听到因反射面和巨大空间所产生的衰减时间很长的声音。该声音当然也可用便携式录音机录制。然后把该脉冲响应(即由击掌时刻开始至 0 幅值时的声音)导入计算机中, 接下来近场录制(close-mike)人声(将传声器贴近嘴巴)并得到一段非常“干”(dry)的人声(房间或空间几乎没有造成影响的声音)。最后可用式 5.6 对两个信号(人声干信号 $x[\cdot]$ 与大教堂的脉冲响应 $h[\cdot]$)进行卷积, 结果得到的声音像是在教堂里听到的一样, 而实际上听者并没有到过该教堂。事实上, 可将任何类型的声音信号作为输入——如以大教堂的脉冲响应对你的一首吉他独奏曲进行卷积, 可得到你梦寐以求的在大教堂里进行吉他独奏的效果。

用如下数列来演示一个卷积实例, $x[n]$ 为输入信号, $h[n]$ 为脉冲响应。

$$x[n] = \{1, 2, 3\} \quad (5.7)$$

$$y[n] = \{4, 5\} \quad (5.8)$$

令式 5.6 中的 $m=0 \cdots 2$, 则卷积输出 $y[n]$ 如式 5.9, 其中 $x[0]$ 之前与 $x[2]$ 之后没有取值, 并包含了所有的 x 与 h 分量(x 取两者中较大的数值)。

$$y[n] = x[n] * h[n] = \sum_{m=0}^{m=2} x[m] \cdot h[n-m] \quad (5.9)$$

此时, 计算该卷积输出的第一项有:

$$y[0] = x[0] \cdot h[0-0] + x[1] \cdot h[0-1] + x[2] \cdot h[0-2] = 1(4) + 0 + 0 = 4 \quad (5.10)$$

随后输出的值为:

$$\begin{aligned} y[1] &= 13 \\ y[2] &= 22 \\ y[3] &= 15 \\ y[4] &= 0 \end{aligned} \quad (5.11)$$

这样的卷积计算过程看起来有些乏味, 不过诸如 C、C++、Java 或 MATLAB[®] 都提供方便的卷积工具包。尽管卷积算法的编程一开始会比较复杂, 但得益于计算机语言所提供的算法代码, 卷积计算也被编入了编程语言中, 因此该计算变得越来越简单。

必须知道的卷积性质

本节将简要讨论卷积的几个重要性质。卷积的交换律是指, 无论哪一个部分被视为脉冲响

应以及哪一个部分被视为没有数学意义上的输入,卷积输出结果保持不变。

$$y[n] = x[n] * h[n] = h[n] * x[n] \quad (5.12)$$

因此无论用 h 对 x 进行卷积还是用 x 对 h 进行卷积,都不影响输出结果。定义交换律后,卷积是线性的,因此要进行卷积,系统必须也是线性时不变的。第8章将在频域重新讨论卷积计算,即卷积对是如何交换的——时域卷积转换到频域时将变为乘法。说到卷积的交换律,不可避免地要提到与之相关的另一个性质,如式 5.13 所示:

$$y[n] = g[n] * (x[n] * h[n]) = (g[n] * h[n]) * x[n] \quad (5.13)$$

由上式可知,利用上式括号部分作为中间过程再与括号外的信号进行卷积,可将两个以上的信号进行卷积。根据线性,卷积的分配律也可如式 5.14 所示:

$$y[n] = g[n] * (x[n] + h[n]) = g[n] * x[n] + g[n] * h[n] \quad (5.14)$$

卷积的另一个重要性质是卷积结果的输出长度由式 5.6 的求和项数控制——进行卷积的两个信号长度分别为 M 和 N 时,卷积后信号的最终长度如式 5.15 所示:

$$convolution_{length} = M + N - 1 \quad (5.15)$$

虽然由式 5.6 中 m 值可知,可对负无穷大至正无穷大的项进行求和,但实际应用中输入与脉冲响应在零值区间“无定义”(幅值等于 0)而没必要取无穷多个求和项。对于上述数列卷积的例子,预计的卷积后信号长度为 $3+2-1=4$,因此,只需要计算 $y[n]$ 的前四项($n=0,1,2,3$)即可。

6. 系统流程图与数字系统框图

如前所述,差分方程由输入信号的加法器、乘法器及其延迟器和输出信号分量组成。图 3.1 和图 3.2 也给出该系统信号流程的非正式的图形化阐释。虽然以传统形式表达的差分方程非常有用,也是长差分方程的缩写形式,但是同一差分方程的图形表达形式有时也是需要的。因为传统形式有点类似基于命令行(command-line-based)的软件而不是以图形用户界面为核心(GUI-centric)的软件,就如计算机音乐语言 Supercollider 之于 Max / MSP 软件一样。命令行形式虽然更简洁,但至少第一眼看上去没有包含必要的信息,而仔细看来,图形化形式往往能更好地表达系统原始总体思路。

无一例外,乘法器、加法器和延迟器都可用框图来表示,并有利于对 LTI 系统的理解和图形化。如图 6.1、图 6.2 和图 6.3 所示,每个图块都有一个或多个输入节点和一个输出节点。

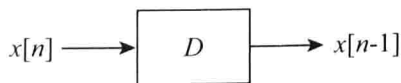


图 6.1 单延迟器

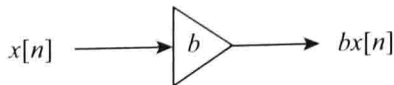


图 6.2 乘法器(增益器)

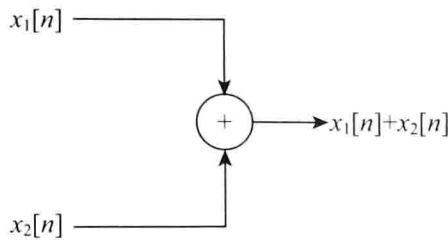


图 6.3 加法器

为了理解以上三图的实际应用,以一个三点移动平均差分方程为例,用系统框图来表达。

$$y[n] = \frac{x[n] + x[n - 1] + x[n - 2]}{3} \tag{6.1}$$

由式 6.1 可知,该方程由两个加法器、两个单位延迟器和一个乘法器组成。因此,其系统框图如图 6.4,其中 $b_0=b_1=b_2=1/3$ 。而式 2.8 包含了反馈分量和非反馈分量的一般差分方程的系统框图,如图 6.5 所示。该图的左半部分表示的是反馈前分量,右半部分表示的是反馈分量,它们都有各自的延迟器和乘法器系数。

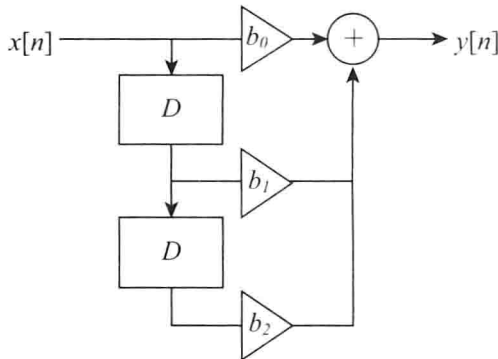


图 6.4 三点移动平均系统框图

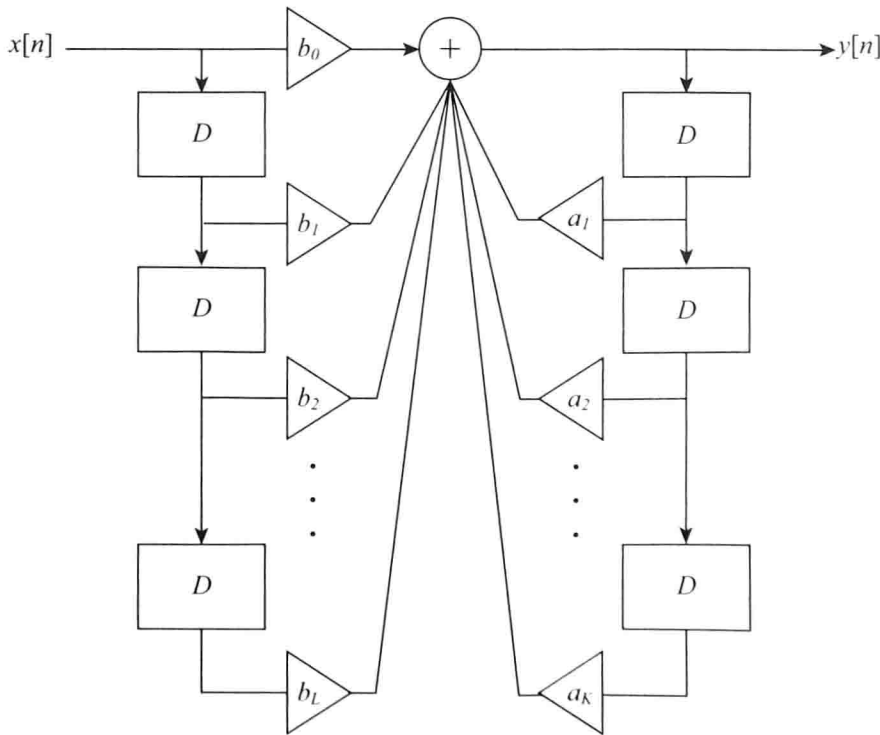


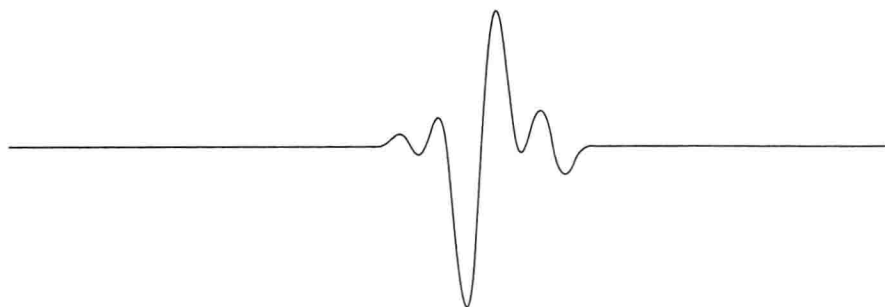
图 6.5 一般差分方程的系统框图

7. 乐例分析

接下来简要介绍卷积在音乐中的运用,此处卷积往往用于给干声信号添加混响。有许多商业应用程序和插件都使用该方法来制造人工混响,而混响器的质量则取决于所录制的脉冲响应的好坏。如有些软件只用一个脉冲响应来模拟整个音乐厅的混响效果,另一些软件则录制了同一个空间不同位置的脉冲响应,通过阶段性地改变混响参数来获得更多更真实的混响特征。由于很容易在互联网上找到来自大厅、录音棚、爵士乐俱乐部、录音棚、建筑空间及其他空间的脉冲响应的录音,卷积很容易在软件中得以实现,并且一旦获得有趣而清晰的脉冲响应,就可让声音片段听起来像是一阵来自于脉冲响应所录制的特定空间的微风。除了制造传统意义上的混响效果外,卷积还可用于其他更有趣的地方,利用非脉冲响应信号进行卷积时,会得到令人惊讶的音色效果。一首这样的作品来自于作曲家特拉维斯·沙尔(Travis Scharr),他在卡里亚娜飓风之后于新奥尔良的超级穹顶体育馆(Superdome)内录制了大量的脉冲响应。这首作品名为《恐惧和希望》(*Of Horror and Hope*, 2007),该作品在超级穹顶体育馆内形象地描绘了飓风来临时的恐怖情景,献给那些飓风袭击期间仍驻守在城市中的人们。2005年,美国一座古老的城市在这场灾难性无法言喻的洪水之中存活了下来。沙尔在该乐曲中利用风暴声、人群呻吟和叫喊声对钢琴声进行卷积。这种作曲手法渲染了一种怪异的洪水音响,该声响后来成为了作品中描述城市重生的“声景”(soundscape)。

另外一个有趣的乐队是名为“卷积兄弟”(convolution brothers)的三人乐队——由在协和声学及音乐联合研究所 IRCAM(Institut de Recherche et Coordination Acoustique / Musique)工作的米勒·帕克特(Miller Puckette)、科特·利珀(Cort Lippe)和萨克·赛特尔(Zack Settel)组成。据笔者所知,他们不仅单独使用基于卷积的效果处理器,而且还联合使用了控制器、计算机音乐编程系统等诸如此类的器件。创作元素包括由特殊软硬件的内外部所产生的素材,其拥有不可思议的实践演出效果。噢,是的,幽默是音乐的重要部分!本章结束之前,需要注意的一个小花絮是:米勒·帕克特就是两个最为流行的计算机音乐系统 Max / MSP 与 Pd(pure data)的创造者,这两个软件利用了系统框图对控制器、效果器、合成器、放大器等数字音频设备或 MIDI 设备进行控制。

第 6 章 频率响应



1. 引 言

第 6 章将围绕“频率响应”(frequency response)的论题展开。频率响应定义为,系统受输入信号激励时,其输出幅值和相位随频率变化的响应特征。输入信号的振幅和相位是常数,但频率却会变化。有些观点认为扬声器和传声器的频率响应有很多种(这一问题还引起过激烈争论)——而这些响应都是指系统的响应。此例中,扬声器本身是一个系统,输入的是一些幅值和相位固定、但频率变化的测试信号。尤其是音乐发烧友经常谈论频率响应非常平直,这指的是扬声器或传声器的性能——即扬声器或传声器能精确而稳定地重放或录制频带宽广的声音,而不发生声染色或失真,也就是指设备反映出的原信号的真实程度。不幸的是,完美的系统是不存在的,每个系统或多或少都会在某些频率上产生某些类型的失真,从而需要由对数频率坐标而不是线性频率坐标来表示的频率响应曲线或“博德图”(Bode diagram)来评价特定系统的性能。

图 1.1 给出了一个典型的幅值频率响应曲线, x 轴是对数频率坐标, y 轴是以 dB 标注的幅值。从该图中可读到许多关于频率响应曲线的有趣问题。

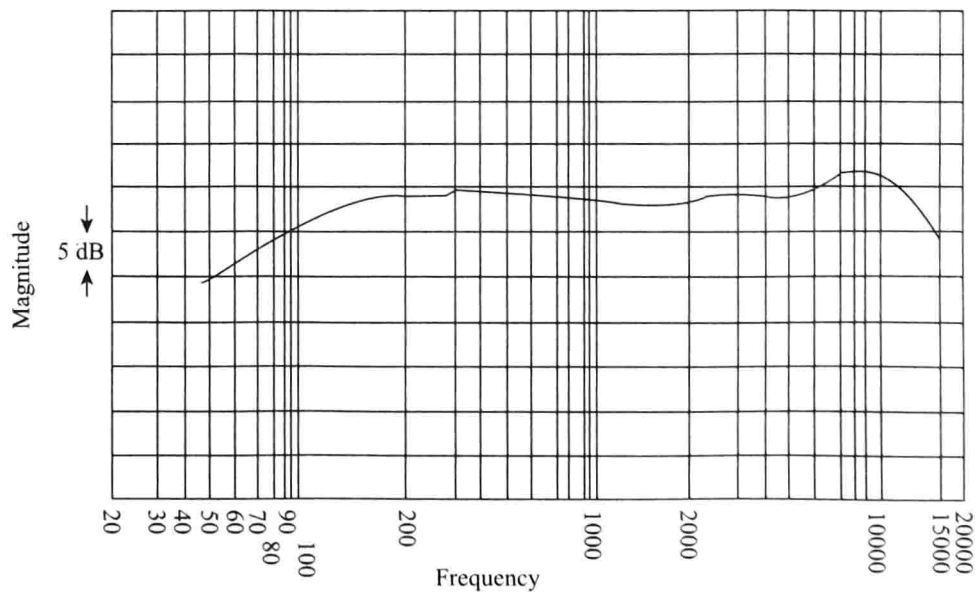


图 1.1 频率响应曲线

2. 频率响应

为了正式定义频率响应,令数字输入信号如式 2.1 所示,幅值为单位增益(由欧拉公式定义的幅值为 1),初相位为 0,数字频率(digital frequency) θ 如式 2.2 所示:

$$x[n] = e^{j\theta n} \quad (2.1)$$

$$\theta = \omega \cdot T = 2 \cdot \pi \cdot f \cdot T \quad (2.2)$$

如果考虑下列情形,则会更好地理解式 2.2 和数字频率的概念。再设将一模拟输入正弦信号 $x(t)$ 输入至一个理想采样器中,如以下式 2.3 所示:

$$x(t) = A \cdot \cos(\omega t + \phi) \quad (2.3)$$

则经过采样后,采样器的输出形式为 $x[n \cdot T]$ (或如第 1 章所述简写为 $x[n]$),对应的模拟输入为 $x(t)$,其中 n 为采样序号, T 为采样周期,单位为秒/采样(seconds/sample)或秒(若为单位采样), ω 为时间连续的角频率,单位为弧度/秒(radians/second), ϕ 为初相位,单位为弧度(radians)。

$$x[nT] = A \cdot \cos(\omega \cdot nT + \phi) \quad (2.4)$$

式 2.4 的缩写形式如式 2.5, θ 为时间离散余弦的数字频率,而 ω 为时间连续的角频率,定义为 $2\pi f$ 。单位为弧度/采样的数字频率 θ 的定义为 $\theta = \omega T = 2\pi \cdot f \cdot T$ 。也就是说 $2\pi \cdot f \cdot T$ 的单位也是弧度/采样,因为(弧度)(赫兹)(秒/采样)=(弧度)(1/秒)(秒/采样点)=弧度/采样:

$$x[n] = A \cdot \cos(\theta \cdot n + \phi) \quad (2.5)$$

接下来如前一章所述,用脉冲响应 $h[m]$ 对输入 $x[n]$ 进行卷积,则有:

$$\begin{aligned} y[n] &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot x[n-m] = \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{j\theta(n-m)} \\ &= e^{j\theta n} \cdot \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j\theta m} = x[n] \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j\theta m} \\ &= x[n] \cdot H(e^{j\theta}) \end{aligned} \quad (2.6)$$

式 2.6 中第四步的求和项即如该式最后一项所示,定义为系统的频率响应,又称为系统的“特征值”(eigenvalue 或 characteristic value)。因此,频率响应的正式定义式为式 2.7:

$$H(e^{j\theta}) = \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j\theta m} \quad (2.7)$$

$$H(e^{j\theta}) = \frac{y[n]}{x[n]} \quad (2.8)$$

由式 2.6 和 2.7 提供的关系可知,如果已知系统的脉冲响应,就能计算系统的频率响应。此外,由观察可知, $H(e^{j\theta})$ 是输出与输入的比值(式 2.8)。一般差分方程如式 2.9,当输入为 $x[n] = e^{j\theta n}$ 时,输出 $y[n]$ 如式 2.10(见式 2.8),则差分方程可改写成式 2.11。

$$y[n] = \sum_{k=1}^{k=N} a_k y[n-k] + \sum_{k=0}^{k=L} b_k x[n-k] \tag{2.9}$$

$$y[n] = x[n] \cdot H(e^{j\theta}) = e^{j\theta n} \cdot H(e^{j\theta}) \tag{2.10}$$

$$e^{j\theta n} \cdot H(e^{j\theta}) = \sum_{k=1}^{k=N} [a_k e^{j\theta(n-k)} \cdot H(e^{j\theta})] + \sum_{k=0}^{k=L} [b_k e^{j\theta(n-k)}] \tag{2.11}$$

式 2.11 又可改写成式 2.12。

$$H(e^{j\theta}) = \frac{\sum_{k=0}^{k=L} b_k e^{-j\theta k}}{1 - \sum_{k=0}^{k=N} a_k e^{-j\theta k}} \tag{2.12}$$

由式 2.12 可知,如果仅知系统的差分方程,将输入频率以单位增益进行变化,再设初相位为 0,也可以直接计算系统的频率响应。式 2.12 的形式显示了频率响应是输入系数和与输出系数和的比值。正如本章开头所说的那样,频率响应是输入信号的振幅和相位不变而频率变化时,输出幅值和相位的变化。因此,频率响应的幅值与相位特征可由式 2.12 计算。代入随其延迟特征变化的特定差分方程系数时,式 2.12 的频率响应的幅值响应可表示为式 2.13。

$$H(e^{j\theta}) = M \cdot e^{j\Theta(\theta)} \tag{2.13}$$

$M(\theta)$ 的幅值,或简写为 M ,有:

$$M = |H(e^{j\theta})| \tag{2.14}$$

相位响应 $\Theta(\theta)$ 为:

$$\Theta(\theta) = \angle H(e^{j\theta}) = \tan^{-1} \left(\frac{\text{imag}(H(e^{j\theta}))}{\text{real}(H(e^{j\theta}))} \right) \tag{2.15}$$

图 1.1 即是输入为 20~20 kHz 的扫频信号时,式 2.14 的图像。

2.1 $H(e^{j\theta})$ 的特征与性质

频率响应存在一些有趣而重要的性质,包括周期性、频率定义域与对称性,如表 2.1 所示。接下来首先讨论频率定义域。

表 2.1 $H(e^{j\theta})$ 的性质

频率定义域	描述 $0 \leq \theta < \pi$
周期性	2π
对称性	幅值响应为偶函数, 相位响应为奇函数

2.1.1 频率域与奈奎斯特定理

第 1 章首次介绍了奈奎斯特定理、采样率和频率混叠的概念。被采样信号的最高频率分量不满足条件 $f_{\max} < f_s/2$ 时,会发生频率混叠,导致高于该较低限制频率的频率分量发生频移并产生人为失真。对于角频率而言, $f_s/2$ 相当于下式 2.19 中的 π ,其中数字频率 θ 的单位通常是弧

度/采样,角频率 ω 的单位为弧度/秒,采样周期 T 的单位为秒/采样,采样率 f_s 的单位为采样数/秒。

$$\theta = \omega \cdot T = 2 \cdot \pi \cdot f \cdot T = 2 \cdot \pi \cdot \frac{f}{f_s} \quad (2.16)$$

或

$$\theta = 2 \cdot \pi \cdot \frac{f}{f_s} \rightarrow f = \theta \cdot \frac{f_s}{2 \cdot \pi} \quad (2.17)$$

由奈奎斯特定理知,该未知最高频率分量必须由式 2.18 加以限制:

$$0 \leq f < \frac{f_s}{2} \quad (2.18)$$

将式 2.18 的中 f 用式 2.17 代入,则以弧度/采样(数字频率)为单位的奈奎斯特限制频率如式 2.19 所示,最后简化如式 2.20。

$$0 \leq \theta \cdot \frac{f_s}{2 \cdot \pi} < \frac{f_s}{2} \quad (2.19)$$

$$0 \leq \theta < \pi \quad (2.20)$$

有了以弧度为单位的采样限制频率将有助于我们更好地处理频率响应。无论一个给定系统的采样频率 f_s 是多少,将 $f_s/2$ 表示为 π 而不是以赫兹为单位的特定频率值,也有助于用优美的欧拉公式来解释频率混叠的发生机制。为了了解频率混叠的发生机制,考察图 2.1,在 DC 直流分量(0 频率)两侧存在镜像对称特征。而在 DC 直流分量至 $\pm\pi$ 之间,对称性似乎更明显。至少由图可知,角频率 π 是指奈奎斯特限制频率,这意味着如果要满足奈奎斯特定理,至少频率响应幅值 $|H|$ 在 $\pi \pm \phi$ 这两个弧度频率上应该是相同的。频率超出 π 后,如 $\pi + \phi$,将会以较低频率 ($\pi - \phi$) 产生如第 1 章 4.2.1 节所述的频率混叠现象。

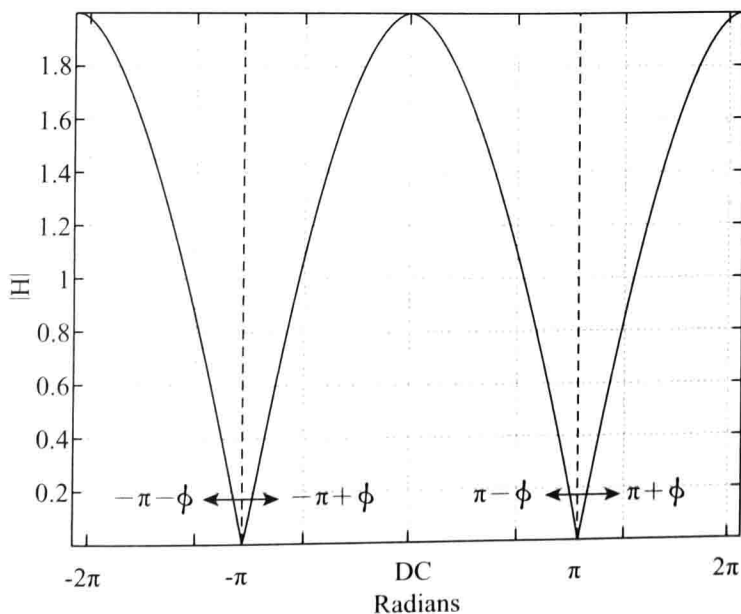


图 2.1 $-2\pi \sim 2\pi$ 的频率响应

当然,对于这个似是而非的现象,以上描述并不充分,但如果这一现象能够得到证明,那么通过频率响应分析就可以揭示频率混叠的发生机制和原因。为了完成该证明,设式 2.21 恒成立。

$$|H(e^{j(\pi+\phi)})| = |H(e^{j(\pi-\phi)})| \quad (2.21)$$

令 $\theta = \pi - \phi$, 由频率响应的定义(式 2.7)开始,则有:

$$\begin{aligned} H(e^{j\theta})|_{\theta=\pi-\phi} &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j\theta m} \\ &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j(\pi-\phi) \cdot m} \\ &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j\pi \cdot m} \cdot e^{j\phi \cdot m} \end{aligned} \quad (2.22)$$

由欧拉公式知 $e^{j\theta} = \cos\theta + j\sin\theta$, 用该式将上式的指数分量展开可知,对于所有整数 m , 正弦分量因值恒为 0 而消失了(π 的整数倍的正弦函数值为 0)。而余弦分量则变成了“极函数”(polarity function), 即对于偶数 m 值为 +1, 对于奇数 m 值为 -1, 如下式所示:

$$e^{-j\pi \cdot m} = \cos(\pi \cdot m) - j\sin(\pi \cdot m) = \cos(\pi \cdot m) - 0 = (-1)^m \quad (2.23)$$

则式 2.22 可展开为:

$$\begin{aligned} H(e^{j\theta})|_{\theta=\pi-\phi} &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot e^{j\phi \cdot m} \\ &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot [\cos(\phi \cdot m) + j\sin(\phi \cdot m)] \end{aligned} \quad (2.24)$$

同理,令 $\theta = \pi + \phi$, 有:

$$\begin{aligned} H(e^{j(\pi+\phi)})|_{\theta=\pi+\phi} &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j(\pi+\phi) \cdot m} \\ &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot e^{-j\pi \cdot m} e^{-j\phi \cdot m} \\ &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot e^{-j\phi \cdot m} \\ &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot [\cos(\phi \cdot m) - j\sin(\phi \cdot m)] \end{aligned} \quad (2.25)$$

现在可知,除了余弦分量与正弦分量之间相差一个负号之外,式 2.24 和 2.25 是完全相同的。如果将方程中所有的实部与虚部都用 $a + jb$ 形式表示,则得到 $H[e^{j(\pi-\phi)}]$ 与 $H[e^{j(\pi+\phi)}]$ 的关系如下:

$$\begin{aligned} H[e^{j(\pi-\phi)}] &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot [\cos(\phi \cdot m) + j\sin(\phi \cdot m)] \\ &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot \cos(\phi \cdot m) \\ &\quad + j \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot \sin(\phi \cdot m) = a + jb \end{aligned} \quad (2.26)$$

$$\begin{aligned}
 a &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot \cos(\phi \cdot m) \\
 b &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot \sin(\phi \cdot m)
 \end{aligned}
 \quad (2.27)$$

同理,对于 $H[e^{j(\pi+\phi)}]$,有:

$$\begin{aligned}
 H[e^{j(\pi+\phi)}] &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot [\cos(\phi \cdot m) + j\sin(\phi \cdot m)] \\
 &= \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot \cos(\phi \cdot m) \\
 &\quad - j \sum_{m=-\infty}^{m=\infty} h[m] \cdot (-1)^m \cdot \sin(\phi \cdot m) = a - jb
 \end{aligned}
 \quad (2.28)$$

最后,这些复杂向量的幅值可用平方和的平方根来计算,如式 2.29:

$$|H| = \sqrt{\text{实部}^2 + \text{虚部}^2} \quad (2.29)$$

则对于角频率 $\pi-\phi$ 和 $\pi+\phi$,其频率响应的幅值响应如下:

$$|H[e^{j(\pi+\phi)}]| = |H[e^{j(\pi-\phi)}]| = \sqrt{a^2 + b^2} \quad (2.30)$$

因此,对于幅值 $|H(e^{j\theta})|$, $\theta=\pi-\phi$ 和 $\theta=\pi+\phi$ 时该值完全相同,同时在直角坐标系中考虑其实部和虚部的极性时,其关于实轴的对称映射如图 2.2。从频域上看,如上述证明所述,超过 π 的频率(y 轴负虚部象限)被对称映射至了正虚部象限,如图 2.3 所示。这就是频率混叠。

第 8 章在介绍了傅立叶变换之后,还会从另一个方面再次讨论频率混叠和奈奎斯特限制频率。

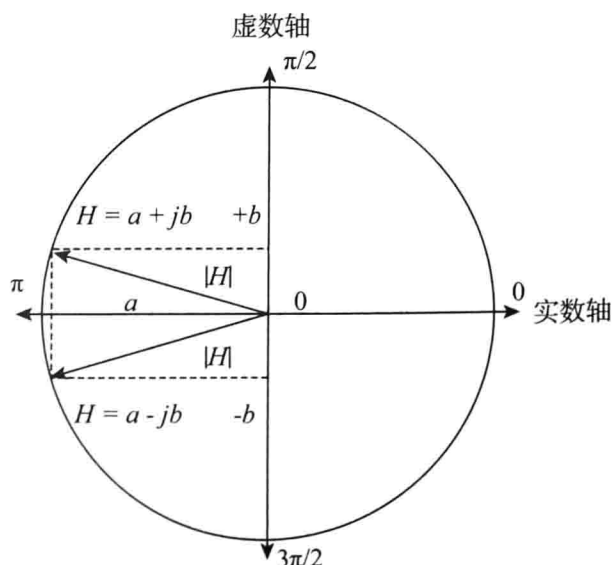


图 2.2 复数直角坐标系上的幅值极性映射

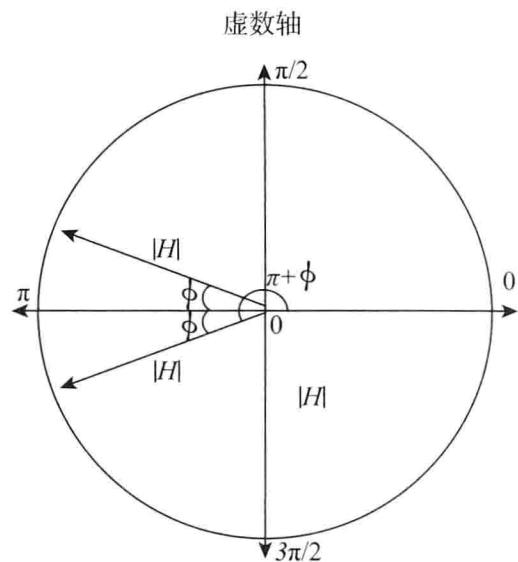


图 2.3 角频率 $\theta=\pi-\phi$ 和 $\theta=\pi+\phi$ 的映射

2.1.2 $H(e^{j\theta})$ 及其周期性

由图 2.2 和图 2.3 已经可以看出 $H(e^{j\theta})$ 的周期性,同时可猜测这种重复特征的来源是欧拉公式中余弦和正弦函数的周期性,即对于所有对应于 $2 \cdot \pi \cdot k$ 的整数 k ,它们都具有周期性。因此,若设式 2.31 恒成立,则频率响应的周期性得证。

$$H(e^{j\theta}) = H[e^{j(\theta+k\pi)}] \tag{2.31}$$

接下来从式 2.31 的右边开始该式的证明:

$$\begin{aligned} H[e^{j(\theta+k\pi)}] &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot e^{-j(\theta+k\pi) \cdot m} \\ &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot e^{-j\theta \cdot m} e^{-jk\pi \cdot m} \end{aligned} \tag{2.32}$$

再由欧拉公式可知,对于所有整数 k ,式 2.32 右边的指数分量与式 2.33 相一致。

$$e^{-jk\pi \cdot m} = \cos(k \cdot 2 \cdot \pi) - j\sin(k \cdot 2 \cdot \pi) = 1 - 0 = 1 \tag{2.33}$$

则式 2.32 可缩写成式 2.34,同时观察该式可知,式 2.31 所描述的周期性恒成立。

$$\begin{aligned} H[e^{j(\theta+k\pi)}] &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot e^{-j\theta \cdot m} \cdot e^{-jk\pi \cdot m} = \sum_{m=-\infty}^{m=+\infty} h[m] \cdot e^{-j\theta \cdot m} \cdot 1 \\ &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot e^{-j\theta \cdot m} \end{aligned} \tag{2.34}$$

2.1.3 对称性

由图 2.1、图 2.2 和图 2.3 可知,频率响应的幅值类似余弦函数关于实轴对称——如果将余弦函数看成初相位为 0 的时间序列,然后在 $x=0$ (时间轴)处增加或减少一个极小相位 ϕ ,幅值将相同且关于 $x=0$ 轴偶对称,而正弦函数则是奇对称。下式可准确表达这种镜像对称性。

$$|H(e^{j\theta})| = |H(e^{-j\theta})| \tag{2.35}$$

至此,使用欧拉公式可简便的证明式 2.35。

$$\begin{aligned} H(e^{j\theta}) &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot e^{-j\theta \cdot m} \\ &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot [\cos(\theta \cdot m) - j\sin(\theta \cdot m)] = c - jd \end{aligned} \quad (2.36)$$

$$\begin{aligned} H(e^{-j\theta}) &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot e^{j\theta \cdot m} \\ &= \sum_{m=-\infty}^{m=+\infty} h[m] \cdot [\cos(\theta \cdot m) + j\sin(\theta \cdot m)] = c + jd \end{aligned} \quad (2.37)$$

再由以上两式可知,两者的频率响应幅值是相同的,如式 2.38 所示,其中对于给定的角频率 ω , c 和 d 为实常数。

$$|H(e^{j\theta})| = |H(e^{-j\theta})| = \sqrt{c^2 + d^2} \quad (2.38)$$

而对于相位响应的对称性,由图 2.2 知呈奇对称,即 $\tan^{-1}(H) = -\tan^{-1}(H)$ 。

$$\angle H(e^{j\theta}) = \tan^{-1}\left(\frac{+b}{a}\right) \quad (2.39)$$

$$\angle H(e^{-j\theta}) = \tan^{-1}\left(\frac{-b}{a}\right) \quad (2.40)$$

$$\angle H(e^{j\theta}) = -\angle H(e^{-j\theta}) \quad (2.41)$$

2.2 频率响应的其他性质

通常有两种绘制频率响应曲线的方法——一种是使用线性幅值,另一种则是使用对数幅值。使用更为标准的对数分贝幅值来表示频率响应的原因之一,是它更具有可操作性。

$$|H(e^{j\theta})|_{\text{dB}} = 20 \cdot \log_{10}(|H(e^{j\theta})|) \quad (2.42)$$

如图 2.4 所示,上图为频率响应的线性幅值,下两图为对数分贝幅值。显然,线性幅值不能很好地显示出与低幅值相关的细节,即线性幅值 y 轴的范围是 0 至 15,而对数幅值的范围扩展到 -10 至 +20。对数分贝幅值曲线很好地表现了同样重要的细微峰值以及低幅值区域的重要细节。另一个选择对数幅值作为标准的原因是,人耳对声音动态的感知是对数的而不是线性的。此外,如本章第 1 节开始所述,博德图也常以对数频率坐标轴来表示频率响应。这样做也是因为人耳的频率感知是对数的——博德图更好地表示了人耳对频率的感知。需要特别注意的是, x 轴以归一化频率表示($\pi \times$ 弧度/采样),它所对应的数字频率形式是 $\theta = \omega T$ 。则当该归一化频率等于 0 弧度/采样时,对应的是 DC 直流分量,等于 π 弧度/采样时,则对应了奈奎斯特限制频率 $f_s/2$ 。

接下来以如式 2.43 的含简单反馈分量的 IIR 系统为例,通过计算频率响应来结束本节的论述,并将新学到的概念运用到实际运算当中。

$$y[n] = x[n] + a \cdot y[n-1] \quad (2.43)$$

用差分方程来计算式 2.12 的频率响应,并注意到其结果包含了一个权重为 a 的反馈分量:

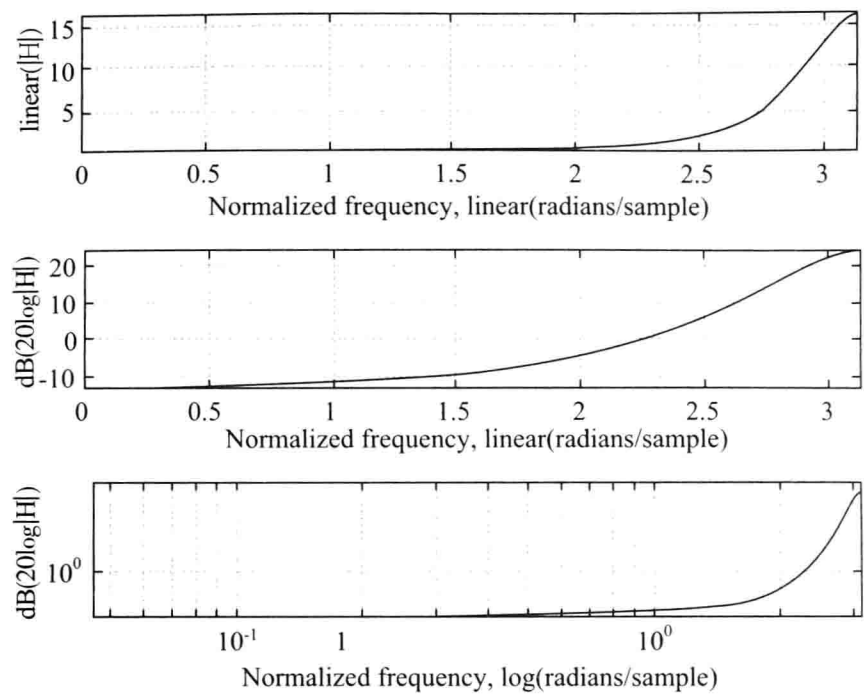


图 2.4 线性及对数博德图表示的频率响应曲线

$$H(e^{j\theta}) = \frac{1}{1 - a \cdot e^{-j\theta}} \tag{2.44}$$

接下来,为了形式上看上去不那么混乱,用虚变量表示分母,即对于给定的数字频率 θ ,可得到如式 2.45 和 2.46 的关系来计算 $|H(e^{j\theta})|$ 。

$$1 - a \cdot e^{j\theta} = 1 - a \cdot (\cos\theta - j\sin\theta) = A + jB \tag{2.45}$$

其中,

$$A = 1 - a \cdot \cos\theta, B = a \cdot \sin\theta \tag{2.46}$$

$$\begin{aligned} |H(e^{j\theta})| &= \left| \frac{1}{A + jB} \right| = \left| \frac{1}{(A + jB)} \cdot \frac{(A - jB)}{(A - jB)} \right| \\ &= \left| \frac{A - jB}{A^2 - B^2} \right| = |(C - jD)| = \sqrt{C^2 + D^2} \end{aligned} \tag{2.47}$$

$$C = \frac{A}{A^2 - B^2} \tag{2.48}$$

$$D = \frac{B}{A^2 - B^2} \tag{2.49}$$

注意到,式 2.47 使用了虚变量来代替分母中别扭的虚部。用极坐标表示频率响应时,有如下结果:

$$H(e^{j\theta}) = Me^{j\Theta(\theta)} \tag{2.47}$$

其中,

$$M = \sqrt{C^2 + D^2} \tag{2.48}$$

$$\Theta(\theta) = \tan^{-1} \frac{D}{C} \quad (2.49)$$

上述所有算法可方便地用诸如 MATLAB® 的编程语言来实现,只要得到上述方程,事情将变得更简单。图 2.5 给出了上例差分方程的线性和对数分贝,其中 $a=0.5$,频率单位为弧度/采样。

这类滤波器称为低通滤波器(low-pass filter),它能够保留低频分量而去除高频分量。本书目录所列的该类滤波器将在下章讨论。

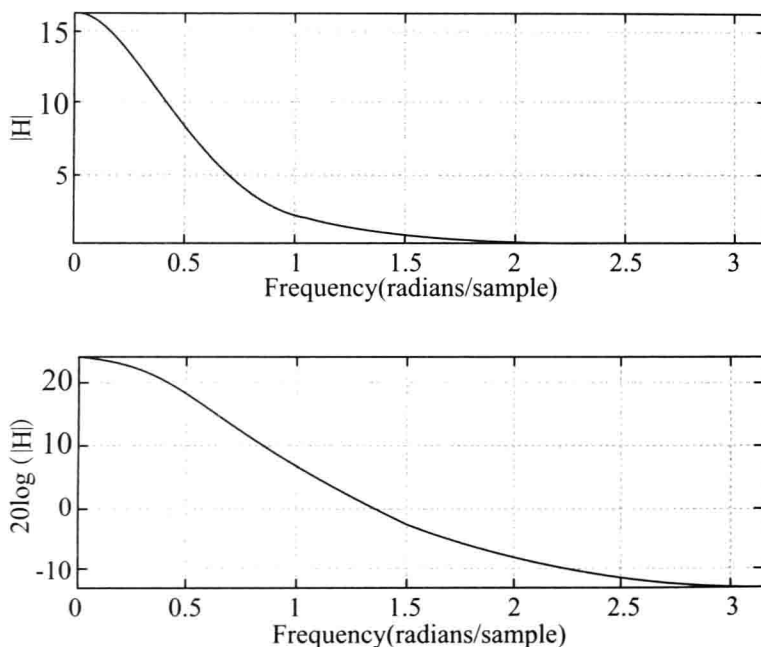


图 2.5 $y[n] = x[n] + 0.5y[n-1]$ 的频率响应

(上图为线性幅值,下图为对数幅值)

3. 相位响应和相位失真

本章第 2 节定义了系统的相位响应 $\Theta(\theta)$,它是传递函数 $H(e^{j\theta})$ 的虚部与实部之比的反正切函数。当系统的输入具有单位增益,初相位为 0 而频率变化时,系统的相位特征如式 3.1 所示:

$$\Theta(\theta) = \angle H(e^{j\theta}) = \tan^{-1} \left(\frac{\text{imag}(H(e^{j\theta}))}{\text{real}(H(e^{j\theta}))} \right) \quad (3.1)$$

第 1 章第 2 节和第 4 章第 2 节还介绍了初相位和瞬时相位的概念,初相位指在 $t=0$ 时刻信号的相位偏移量,而瞬时相位指随时间变化的相位。

如式 3.1 所示的相位响应指随输入频率变化的输出信号的相移。如果向系统 $H(\cdot)$ 输入一个余弦信号,则其输出不仅在振幅响应上有变化,而且在相位响应上也会有变化——相位的不同变化取决于余弦信号的频率。也可从另一个更直观的层面理解相位和相位响应,即输入信号经过系统 $H(\cdot)$ 时发生延迟而产生的相位。例如,将你的电吉他(输入信号)通过声卡(设延迟为 0,对信号没有影响)接入计算机,然后使用软件(系统 H)对该信号添加失真效果。理想

情况下,输出信号只有吉米·亨德里克斯(Jimmy Hendrix)失真效果而没有延迟。也就是说,如果由于软件本身的任何原因,信号延迟足够大,如达到 1 秒左右,则你无法用手进行实时演奏,其效果就像是有一台计算机在舞台上演奏一样酷了。而如果延迟足够短,演奏就可与其他演奏家同步,当然也包括你自己。更令人感兴趣的情况是,相位的变化取决于演奏者所演奏的音高——音高越高延迟越长,音高越低则延迟越短。而由软件延迟的模式所产生的“传播时间”(propagation time)就与本节将要讨论的相位响应息息相关。

图 3.1 显示了如式 3.2 所示的简单无反馈系统差分方程的幅值响应和相位响应——这个系统是一个简单的 FIR 低通滤波器。

$$y[n] = x[n] + x[n-1] \quad (3.2)$$

像移动平均滤波器一样,如果将输入分量的和除以 2(即 $x[n]/2 + x[n-1]/2$)或将 $x[n]$ 与 $x[n-1]$ 的和乘以 0.5,将得到一个两点算术均值的差分方程。如前所述,此例与前例的 IIR 差分方程一样也是一个低通滤波器——即所有的平均滤波器都是低通滤波器。下一章会特别针对音乐,来进一步讨论最常见的滤波器类型及其令人感兴趣的应用。

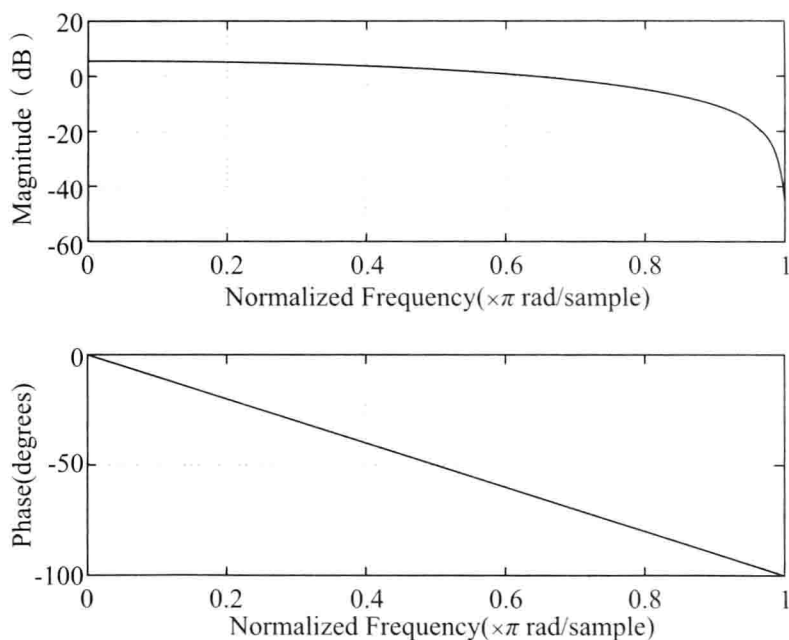


图 3.1 系统的幅值响应与相位响应

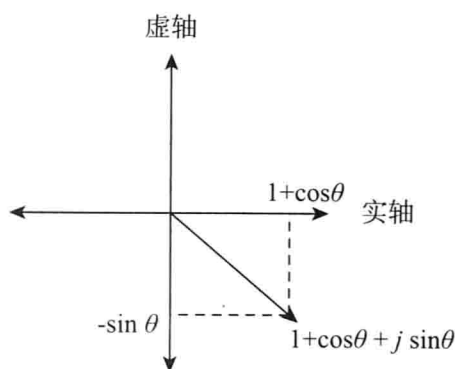
要分析这个系统,首先用式 2.12 将该差分方程改写成频率响应形式,如下式 3.3 所示:

$$H(e^{j\theta}) = 1 + e^{-j\theta} = 1 + \cos\theta - j\sin\theta \quad (3.3)$$

由式 3.3 可知其实部和虚部,如图 3.2 所示,由该图知可用式 3.1 计算其相位响应。

$$\begin{aligned} \Theta(\theta) &= \tan^{-1} \left(\frac{\text{imag}(H(e^{j\theta}))}{\text{real}(H(e^{j\theta}))} \right) \\ &= \tan^{-1} \left(\frac{-\sin\theta}{(1 + \cos\theta)} \right) \end{aligned} \quad (3.4)$$

利用式 3.5 和 3.6 的基本三角公式,相位响应可简化为式 3.7。

图 3.2 $H(e^{j\theta})$ 的复数表示

$$\cos(2 \cdot \theta) = 2 \cdot \cos^2 \theta - 1 \leftrightarrow \cos(\theta) = 2 \cdot \cos^2\left(\frac{\theta}{2}\right) - 1 \quad (3.5)$$

$$\sin(2 \cdot \theta) = 2 \cdot \sin \theta \cdot \cos \theta \leftrightarrow \sin(\theta) = 2 \cdot \sin\left(\frac{\theta}{2}\right) \cdot \cos\left(\frac{\theta}{2}\right) \quad (3.6)$$

$$\begin{aligned} \Theta(\theta) &= \tan^{-1}\left(\frac{-\sin \theta}{1 + \cos \theta}\right) = \tan^{-1}\left(\frac{-\sin(\theta/2) \cdot (2 \cdot \cos(\theta/2))}{2 \cdot \cos^2(\theta/2)}\right) \\ &= \tan^{-1}\left(\frac{-\sin(\theta/2)}{\cos(\theta/2)}\right) = \tan^{-1}(\tan(-\theta/2)) = -\theta/2 \end{aligned} \quad (3.7)$$

再由式 2.2 知 $\theta = \omega T$, 则式 3.7 可以由 $2\pi f$ 定义的时间连续角频率 ω 改写为:

$$\Theta(\theta) = -\frac{\theta}{2} = -\frac{\omega \cdot T}{2} \quad (3.8)$$

同样由图 3.1 可知, 相位响应是线性的, 则以采样点或秒计算的相位延迟为常数(详见下节)。也就是说, 输出端的输出本身是一个非时变波形, 即输出是高频衰减的输入的延迟信号。仅观察图 3.1 的下图, 这种线性相位特性并不明显。但如果将 x 轴的单位换成弧度/采样时, 可知线性相位实际上表示了一个采样周期内, 对于一个特定频率产生多少“弧度距离”(radian distance), 即“相位漂移”(traveled phase)。如频率较低的 1Hz 信号将可用更多的采样点(恰等于采样率 f_s)对一个完整周期进行采样, 反之亦然, 则 1Hz 信号在一个采样周期内不会产生过多的相位漂移。反之对于高频信号——每秒钟的分辨率更高, 则沿相位轴的位移量比低频分量大。这也解释了图 3.1 的下图中, 频率越高相位响应越大, 频率越低相位响应越小的原因。也可将 y 轴表示为弧度, 而不是角度。下式是角度与弧度之间的转换关系:

$$\text{角度} = \text{弧度} \frac{180}{\pi} \quad (3.9)$$

此例最重要的特征, 是这一特定系统所表现出的线性相位响应。在简要回顾相位响应的线性特征之前, 为便于更直观地了解相位, 下一节将先讨论“相位延迟”(phase delay)。

3.1 相位延迟

有时候, 观察采样延时或时延, 比图 3.1 中通过角度或弧度表示的相位图更便于直观地反映出系统的相位响应。就像一个余弦波可看成在时间轴上平移 $\pi/2$ 的正弦波(对于 1Hz 的正

弦或余弦波, $\pi/2$ 弧度等于 1/4 秒), 即用采样数/时间 (samples/time) 而不是相位角或弧度来表示相位响应——这一表示方法称为相位延迟 (phase delay)。因此, 对于相位延迟, 目的在于使输出相位的单位为采样数 (或乘以系数 $T=1/f_s$ 的时间), 如式 3.10 所示:

$$\tau_p(\theta) = -\frac{\Theta(\theta)}{\theta} \tag{3.10}$$

因此, 对于式 3.2 的简单 FIR 低通差分方程, 相位延迟为半个采样周期的时延, 如式 3.11 和图 3.3 所示:

$$\tau_p(\theta) = -\frac{\Theta(\theta)}{\theta} = -\frac{-\frac{\theta}{2}}{\theta} = \frac{1}{2} \tag{3.11}$$

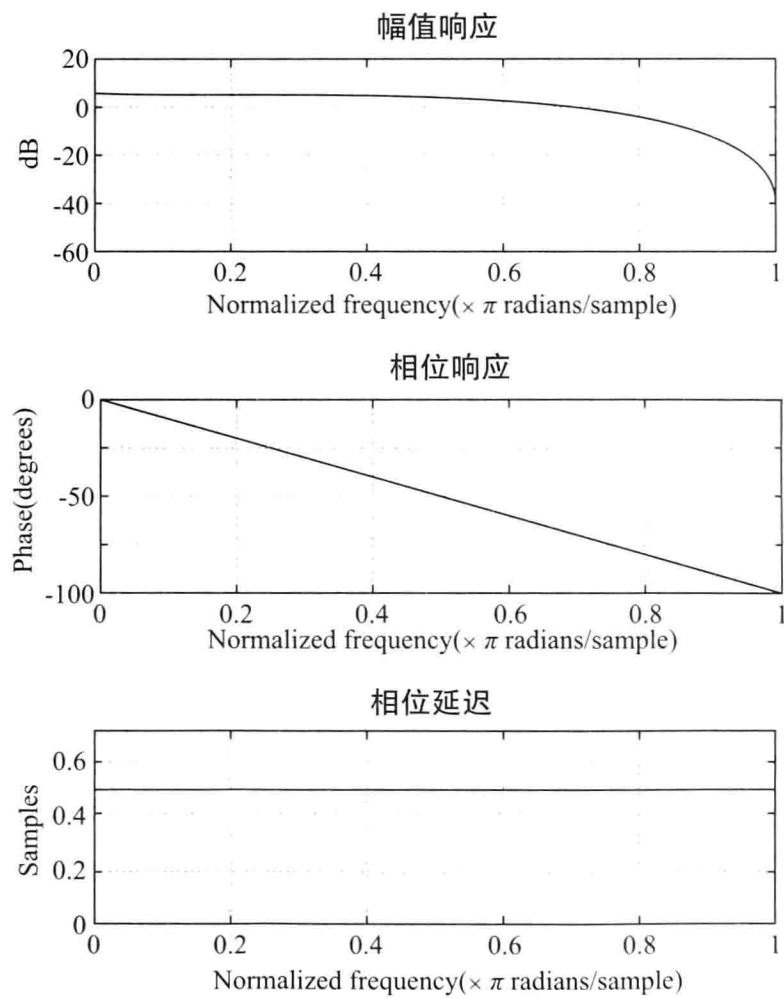


图 3.3 系统 $y[n] = x[n] + x[n-1]$ 的相位延迟
(上图为幅值响应, 中图为相位响应, 下图为采样输出的相位延迟)

对于此例的 FIR 系统, 无论何时将输入信号加入具有这一特定传递函数 $H(\cdot)$ 或差分方程的系统, 其相位延迟恒为 1/2 个采样周期, 而不需要考虑输入信号的频率。如果要以时间而不是采样数来表示相位延迟, 则可如式 3.12 所示乘以采样周期 $T=1/f_s$ 。对于采样率 $f_s=8\text{ kHz}$ 或采样周期 $T=1/8000=0.125\text{ 毫秒}$ 的系统, 相位延迟量将是一个常数 $\tau_p=0.000125/2=0.0625\text{ 毫秒}$ 。这一延迟看起来还不那么糟糕!

$$t_p(\theta) = \tau_p(\theta) \cdot T = \tau_p(\theta) \cdot \frac{1}{f_s} \quad (3.12)$$

相位延迟的另一个定义是,初相位为 ϕ (单位为弧度)的正弦函数等价于其输出的相位延迟为 t_p (单位为秒)的正弦函数,如式 3.13 (优化自 Claerbout 1998)。

$$\sin(\omega t - \phi) = \sin\omega(t - t_p) \quad (3.13)$$

式 3.13 右边括号内的表达式可进一步简化为式 3.14,求解 t_p 得式 3.15。

$$\omega t - \phi = \omega(t - t_p) = \omega t - \omega t_p \quad (3.14)$$

$$t_p = \frac{\phi}{\omega} \quad (3.15)$$

比较式 3.10 和 3.15 可知,一个是时间连续形式(时间的单位为秒,具有连续的角频率),另一个是时间离散形式(时间的单位为采样数,具有离散的数字角频率),时间离散形式的相位延迟多了一个负号。

$$t_p(\omega) = + \frac{\phi}{\omega} \leftrightarrow \tau_p(\theta) = - \frac{\Theta(\theta)}{\theta} \quad (3.16)$$

3.2 线性与相位

具有线性相位响应的系统,其输出相位延迟在整个频率范围内是一个常数。上述 FIR 系统就是这样一个相位延迟恒为半个采样周期的滤波器。需要指出的是,通过单位为采样/时间而不是角度或弧度的延迟,也许可以更直观地观察相位特征。

不过,也许令人惊讶的是,有些传递函数并不能表现出线性的相位响应。作用于输入的相位影响程度取决输入信号的频率。特别是相位特征随频率显著变化时,非线性相位是不需要的。考虑以下极端情况可更好地理解保持相位线性特征的重要性。如一张音频 CD 中有一个由电贝斯、短笛和 64 键电钢琴组成的三人乐队所演奏的音轨,其中电贝斯演奏的频域低于 100Hz,64 键电钢琴在 100Hz 到 600Hz 之间,短笛高于 600Hz,则 CD 播放机播放该音轨时,三件乐器的声音在时间上应该如在一起演奏或在录音棚录制时一样准确(设三人在录制乐曲时的演奏有着极好的节奏和音准——事实上,的确有一支名为 Casiopea 的融合爵士乐队,在 20 世纪 80 年代发行的名为《完美生活》(Perfect Live)的专辑。虽然他们穿着像 80 岁街头艺人埃斯·富兰霍利(Ace Frehley)式的服装,又唱又跳,将演奏水平发挥到了极致,但他们的乐曲听起来仍像机械的 MIDI 序列声。而现在有一台家庭自制的 CD 播放机,声音风格有点放克(funk)风格,相位响应也很有趣——低于 100Hz 的频率相位延迟 2 秒,100Hz 至 600Hz 无延迟,600Hz 以上有 1 秒的延迟。当用这台特殊的播放机播放上述三人乐队的乐曲时,声音非常奇特而有趣——三件乐器似乎根本不是同时演奏的,声音并不同步,如图 3.4 所示:

上例就是由非线性相位响应系统所引起的不需要的信号时域失真。如果使用上述非线性相位系统,输出信号的波形也会随输入频率分量产生不同的相位延迟。图 3.5 给出了具有非线性相位特征的信号波形,其中上图为线性相位,中图为非线性相位,下图为两者的误差。由图可知,非线性相位失真给信号带来了轻微的误差。非线性相位均衡器(nonlinear phase equalizer)通常用在母带处理时按需要对信号做尽可能细微的调整。一种分析相位的非线性特征的方法

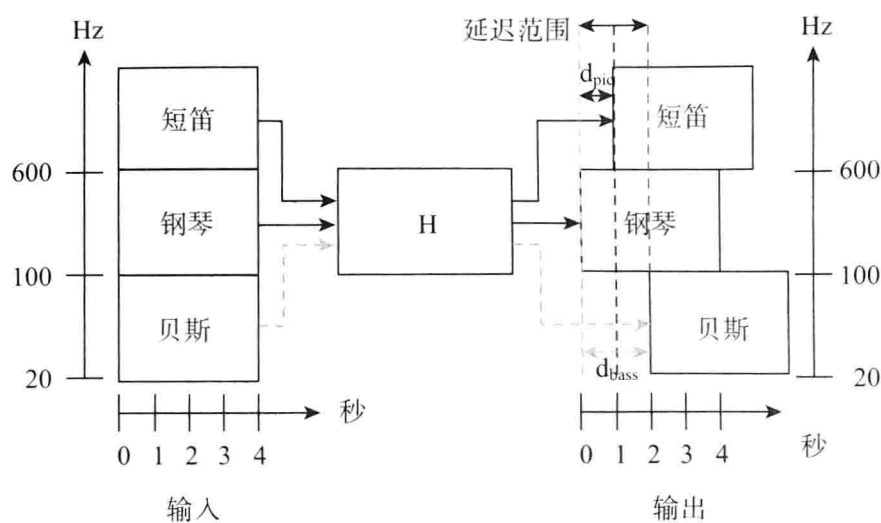


图 3.4 三人乐队中的电贝司、电钢琴和短笛通过
传递函数为 H 的 CD 播放器播放时具有不同的相位延迟

称为“群延迟”(group delay), 详见第 3.3 和 3.4 节。

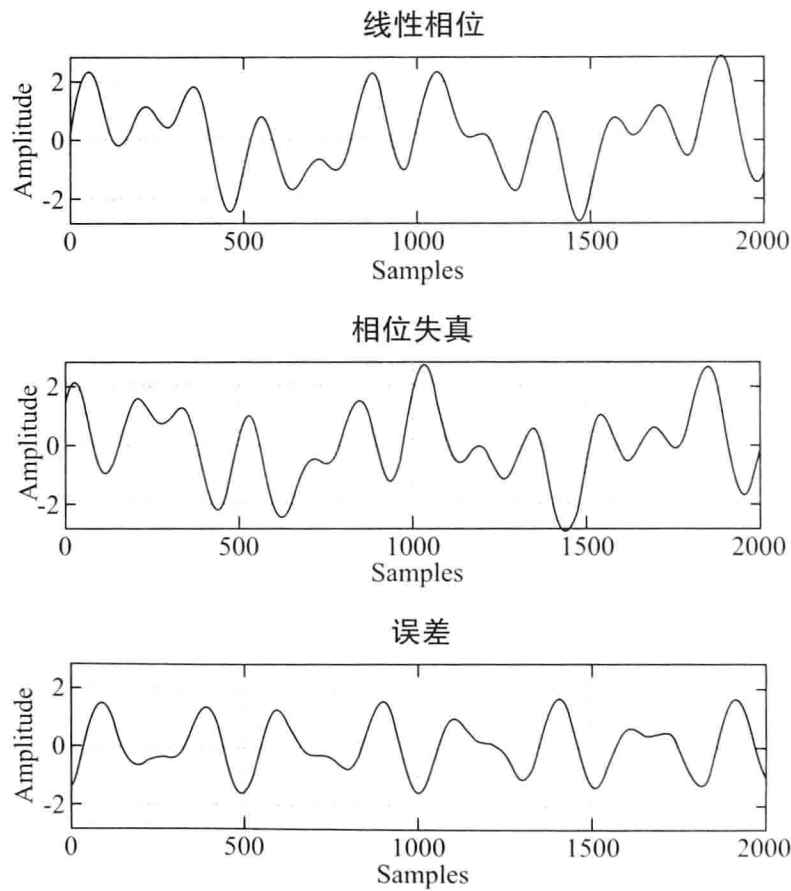


图 3.5 线性相位(上图)与非线性相位(中图)及其误差(下图)

3.3 相位响应与连续相位

计算相位响应时,有时会碰到相位响应图沿频率轴突变,并跳至较远的相位值,如图3.6。仔细观察该相位响应会注意到,如果将相位响应图的后部(约位于 0.45π 弧度)向下平移 $-\pi$ 弧度,则相位响应曲线将变成频率的连续函数,这种方法称为“相位展开”(phase unwrapping)。

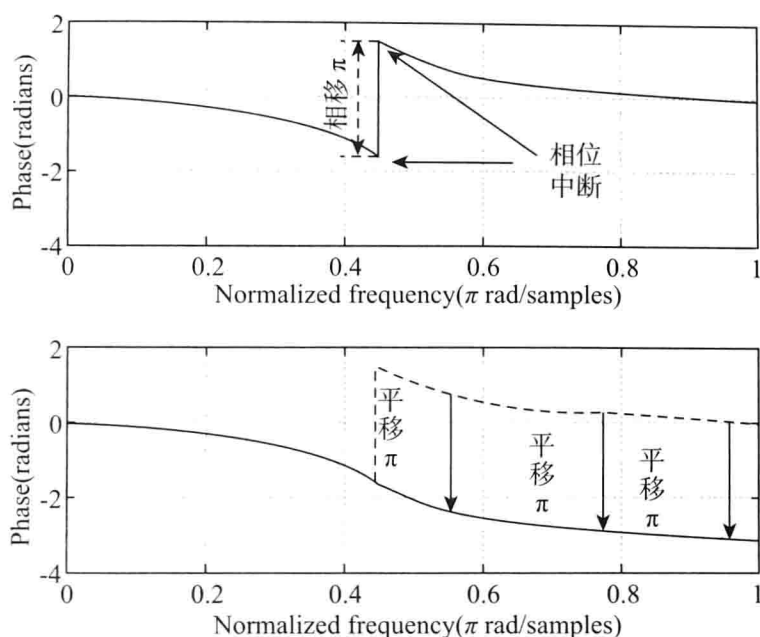


图 3.6 非连续相位与相位偏移 (此例相位偏移量为 π)

这种相位突变是由正弦或余弦函数的周期模数特征所产生的。相位超过 2π 弧度点时将重置为 0——即周期模数为 2π (360 度)。这就如同 12 小时制的时钟,以 12 为模数的十二平均律,又或者是以 12 为模数的年历法。

3.4 群延迟

群延迟 (group delay, 在波动传播理论中对应组速度的时延差) 的概念能够很好地用于分析非线性的相位响应特征。如前一节所述,相位响应与输入频率有关,而相位响应的非线性会造成相位失真和时域波形失真。例如,群延迟是扬声器设计的重要环节,在其重放信号时必定会添加群延迟。对于不同的频率,可测量的相位延迟也会不同。而人耳不会对所有频率同样敏感,对于不同的频率,群延迟也存在不同的阈值。因此,一个高品质的声音重放系统设计团队在设计和优化调整系统的群延迟特征时,会在诸多事项中同时考虑群延迟的阈值特性。对于群延迟失真及其感知,有趣的是,它不仅是频率函数的阈值 (即群延时曲线),也会受信号类型和听众的训练背景 (即对信号的熟悉程度) 的影响 (Blauert, Laws 1978)。

现在介绍群延迟的定义和数学表达式,如式 3.17,它是相位响应与数字角频率的导数比,单位为采样数。

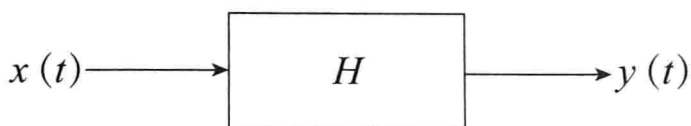
$$\tau_g(\theta) = -\frac{d\Theta(\theta)}{d\theta} \quad (3.17)$$

若相位响应输出是线性的,或可用式 3.18 的形式表示 (如 FIR 系统 $y[n] = x[n] + x[n-1]$),则群延迟与相位延迟相等。

$$\Theta(\theta) = \alpha \cdot \theta \quad (3.18)$$

群延迟一词可能会有些误导,因为它不是指系统的平均延迟,而是通过计算相位响应的斜率来反映其非线性特征。若考虑一对余弦波经过一个未知系统 $H(\cdot)$ 后的输出,如式 3.19 和图 3.7 (优化自 Claerbout 1998),可能会对群延迟的基本概念有更为深入的理解。

$$x(t) = \cos(\omega_1 t) + \cos(\omega_2 t) \quad (3.19)$$

图 3.7 系统 H 的输入与输出

如果系统输出 $y(t)$ 的结果不随两个输入余弦波的频率或振幅分量变化,而只加入了不同的相位延迟量,则输出 $y(t)$ 可改写成式 3.20:

$$y(t) = \cos(\omega_1 t - \phi_1) + \cos(\omega_2 t - \phi_2) \quad (3.20)$$

由第 4 章第 4 节的三角公式可知,式 3.19 的结果如式 3.21 的和差形式,其中频率差通常指人耳能感知到的拍频,这反过来也对应着信号的振幅包络。同理,利用三角公式将式 3.20 写成“包络乘积信号”(envelope multiplying the signal)形式可得式 3.22。

$$x(t) = 2 \cdot \cos\left(\frac{\omega_1 - \omega_2}{2}t\right) \cdot \cos\left(\frac{\omega_1 + \omega_2}{2}t\right) \quad (3.21)$$

$$y(t) = 2 \cdot \cos\left(\frac{\omega_1 - \omega_2}{2}t - \frac{\phi_1 - \phi_2}{2}\right) \cdot \cos\left(\frac{\omega_1 + \omega_2}{2}t - \frac{\phi_1 + \phi_2}{2}\right) \quad (3.22)$$

观察式 3.22 的振幅包络分量(即拍频分量:频率差),用时延和相位延迟改写如下式。同理,由式 3.23 开始可得到相位延迟。式 3.23 左边显示了式 3.22 包络分量的相位延迟,单位为弧度,而右边则显示了式 3.22 包络分量的时间偏移量 t_g 。

$$\cos\left(\frac{\omega_1 - \omega_2}{2}t - \frac{\phi_1 - \phi_2}{2}\right) = \cos\left[\frac{\omega_1 - \omega_2}{2}(t - t_g)\right] \quad (3.23)$$

对比上式两侧括号中的变量,有以下几个关系:

$$\left(\frac{\omega_1 - \omega_2}{2}t - \frac{\phi_1 - \phi_2}{2}\right) = \left(\frac{\omega_1 - \omega_2}{2}t - \frac{\omega_1 - \omega_2}{2}t_g\right) \quad (3.24)$$

$$-\frac{\phi_1 - \phi_2}{2} = -\frac{\omega_1 - \omega_2}{2}t_g \quad (3.25)$$

$$\phi_1 - \phi_2 = (\omega_1 - \omega_2)t_g \quad (3.26)$$

$$\frac{\phi_1 - \phi_2}{\omega_1 - \omega_2} = t_g \quad (3.27)$$

因此由式 3.27 可知,分母和分子分别对应于频率和相位变化,则式 3.27 可写成 3.28:

$$t_g = \frac{\phi_1 - \phi_2}{\omega_1 - \omega_2} = \frac{\Delta\phi}{\Delta\omega} \quad (3.28)$$

式 3.28 的连续形式可写为:

$$t_g = \frac{\phi_1 - \phi_2}{\omega_1 - \omega_2} = \frac{d\phi}{d\omega} \quad (3.29)$$

则数字频率的群延迟为式 3.30 所示,正是本节开始时对群延迟的定义——式 3.17。

$$\tau_g = - \frac{d\Theta}{d\theta} \tag{3.30}$$

图 3.8 给出了一个带阻滤波器的幅值/相位响应、相位延迟与群延迟——带阻滤波器是一种阻止某个已选定频段能量的滤波器(下章将正式学习滤波器)。该系统的差分方程与图 3.5 相同,其中为了使曲线连续跳跃的相位已经展开。由图可知,相位响应是非线性的而不是一条直线(图中上部的第二图),原因是其相位延迟和群延迟不一样。由各曲线的变化可知,系统的群延迟约在 0.5 至 3.5 个采样点之间变化,最大变化发生在如图 3.8 下图所示的 0.45 弧度附近。

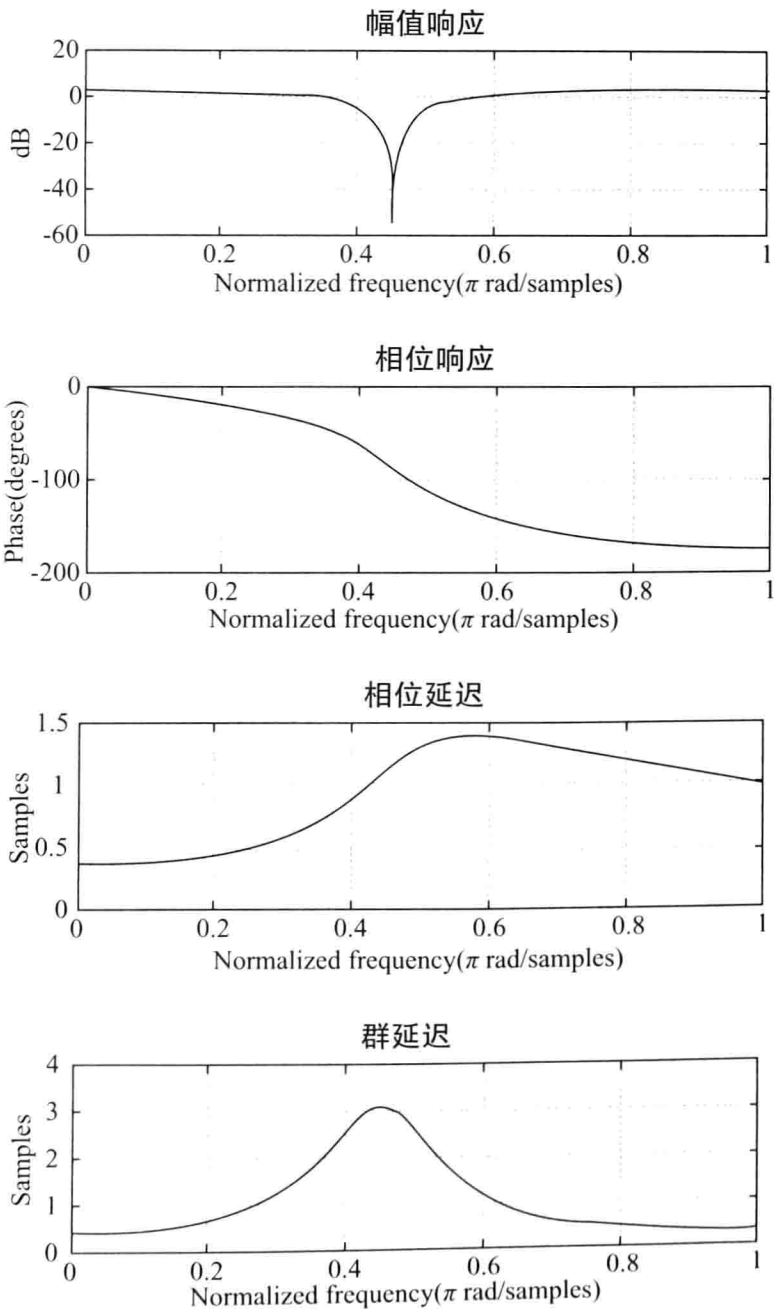


图 3.8 带阻滤波器的幅值响应、相位响应、相位延迟与群延迟
(上图为幅值响应,第二图为相位响应,第三图为相位延迟,下图为群延迟)

4. 神奇的 Z 变换

上一节已经介绍了计算 LTI 系统频率响应的方法,计算过程也许不是最复杂的,但似乎也不是最容易的。本节将介绍另一种计算频率响应的方法,称为“Z 变换”(z-transform)。为什么要使用另一种方法呢?其中的一个原因是,它是一种简单、快捷而且也较为方便的频率响应计算方法。我们由 z 变换的定义开始介绍这一方法,如式 4.1 和 4.2,其中 $x[n]$ 是 z 变换过程的输入信号, $X(z)$ 是输出信号。这两个如此简单的式子可能会让读者感到惊讶。为了解答读者的“啊,这个式子是什么意思”的疑问,接下来就将介绍一些更重要的概念,同时会让读者的惊讶变成“惊叹”。

$$x[n] \xrightarrow{\text{z变换}} X(z) \quad (4.1)$$

$$Z(x[n]) = X(z) = \sum_{n=-\infty}^{+\infty} x[n] z^{-n} \quad (4.2)$$

Z 变换的一个重要性质如式 4.3 所示,称为“可转移性”(shifting property),指存在延迟的输入信号 $x[n]$ 及其对 z 变换的影响,此例的延迟为 m 个采样点。

$$Z(x[n-m]) = \sum_{n=-\infty}^{+\infty} x[n-m] z^{-n} \quad (4.3)$$

令 $k=n-m \rightarrow n=k+m$,并代入式 4.3,得:

$$\begin{aligned} \sum_{n=-\infty}^{+\infty} x[n-m] z^{-n} \Big|_{n=k+m} &= \sum_{k=-\infty}^{+\infty} x[k] z^{-(k+m)} \\ &= \sum_{k=-\infty}^{+\infty} x[k] z^{-k} z^{-m} \\ &= z^{-m} \sum_{k=-\infty}^{+\infty} x[k] z^{-k} \end{aligned} \quad (4.4)$$

注意式 4.4 末尾的求和部分与 $Z(x[n])$ 的形式相同,差别只在于变量的名称: $n=k$ 。因此,通过从 k 到 n 的变量代换,式 4.4 可改写成下式:

$$x[n-m] \xrightarrow{\text{z变换}} z^{-m} X(z) \quad (4.5)$$

$$Z(x[n-m]) = z^{-m} \sum_{n=-\infty}^{+\infty} x[n] z^{-n} = z^{-m} X(z) \quad (4.6)$$

这个有趣的结果说明,z 变换中有平移量(延迟)时,该平移量可以通过 z 的幂的形式转移到 z 变换求和的外部(乘法的系数),结果如式 4.5 和 4.6 所示。到目前为止,这一性质对频率响应的计算有何帮助呢?继续读下去就能找到更多的答案。

4.1 Z 变换的含义(1)

本节将介绍如何将一般差分方程的 z 变换,通过 z 平面(z-plane)的“极点”(pole)和“零点”(zero)来描述系统的频率响应。接下来求解式 4.7 差分方程的 z 变换。z 变换的固有机

制(很多时候会使用大写或其他变量名)涉及输入、输出信号及其时延分量,如式 4.8 和式 4.9 所示。

$$y[n] = \sum_{k=1}^{k=N} a_k y[n-k] + \sum_{k=0}^{k=L} b_k x[n-k] \quad (4.7)$$

$$x[n-m] \rightarrow z^{-m} \cdot X(z) \quad (4.8)$$

$$y[n-m] \rightarrow z^{-m} \cdot Y(z) \quad (4.9)$$

一般差分方程通过 z 变换后可得式 4.10 和 4.11:

$$Y(z) = Y(z) \sum_{k=1}^{k=N} a_k z^{-k} + X(z) \sum_{k=0}^{k=L} b_k z^{-k} \quad (4.10)$$

$$Y(z) \left(1 - \sum_{k=1}^{k=N} a_k z^{-k}\right) = X(z) \sum_{k=0}^{k=L} b_k z^{-k} \quad (4.11)$$

将式 4.11 改写成 $Y(z)/X(z)$ 的形式(传递函数),得式 4.12:

$$\frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{k=L} b_k z^{-k}}{1 - \sum_{k=1}^{k=N} a_k z^{-k}} \quad (4.12)$$

该式看起来非常熟悉,将它与式 2.12 频率响应的定义比较可知,令 $z = e^{j\theta}$,则式 4.12 与式 2.12 的形式完全相同:

$$H(e^{j\theta}) = H(z) \big|_{z=e^{j\theta}} \quad (4.13)$$

虽然从差分方程转化为频率响应的 z 变换 $H(z)$ (即频率响应)本身很直观,但这并不意味着马上就能够用这一结果来计算频率响应。而频率响应的计算需要用到下一节所介绍的知识:极点和零点。

4.2 Z 变换的含义(2):极点和零点

本节将讨论获得 z 变换极点和零点的方法,以及如何使用极零点来计算系统的频率响应。极点和零点可通过求解 $H(z)$ 的分母与分子的根而得到——令分子和分母等于 0 (设它们带有延迟分量),然后以 z 为未知量找到分子和分母的解。通过求解分子和分母的根,找到对应位置的极点和零点,就可快速确定系统的频率响应特性,因为极零点直接反映了系统的共振和阻尼特征,如图 4.1 所示。

$$H(z) = \frac{\text{分子}(z)}{\text{分母}(z)} \quad (4.14)$$

$$\text{分子}(z) = 0 \quad (4.15)$$

$$\text{分母}(z) = 0 \quad (4.16)$$

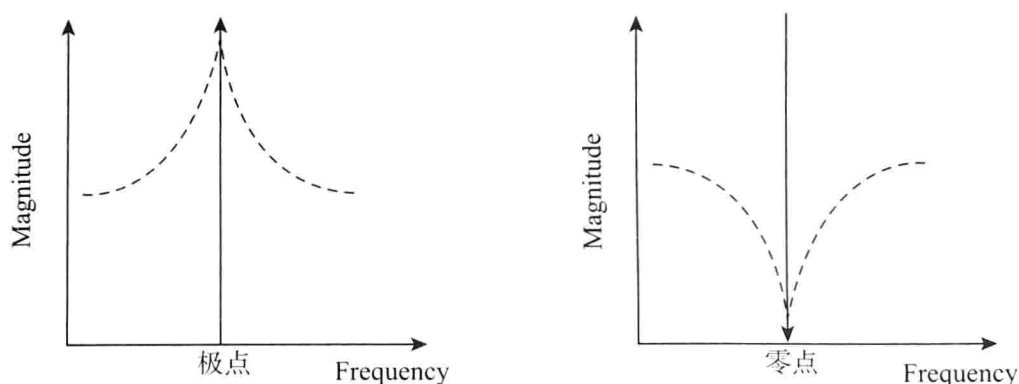


图 4.1 极点与零点的推挽特征

共振位置是指一个由极点定义的、能够增强系统幅值的特殊频点,阻尼特征则是指由一般由零点定义的、能够抑制或降低系统幅值的频点。共振和阻尼特征可以直观地用极点和零点来理解——极点对应着分母的根,则传递函数 $H(z)$ 将变成 $Y(z)/0$ 的形式,被 0 除时 $H(z)$ 将变为无穷大。对于幅值响应曲线,就好像在极点位置有一顶帐篷撑起幅值的响应包络,如图 4.1 中的左图。反之,零点位置,传递函数 $H(z)$ 可写成 $0/X(z)$ 的形式而使 $H(z)=0$,就好像通过帐篷挂钩将帐篷拉到 0 或地面一样。请继续读下去,以了解更多关于幅值响应 $|H|$ 的推挽特征的内容。

4.3 Z 变换的含义(3):单位圆与 Z 平面

为了更好地理解与幅值响应形式有关的极零点的含义,我们以差分方程为 $y[n] = x[n] + 0.5y[n-1]$ 的 IIR 系统为例,通过 z 变换来计算其频率响应。首先计算 $H(z)$ 得:

$$H(z) = \frac{1}{1 - 0.5 \cdot z^{-1}} = \frac{z}{z} \cdot \frac{1}{1 - 0.5 \cdot z^{-1}} = \frac{z}{z - 0.5} \quad (4.17)$$

为了求解分母和分子的根,令这两个部分等于 0,可得以下两个方程,从而求出极点和零点。

$$z - 0.5 = 0 \quad (4.18)$$

$$z = 0 \quad (4.19)$$

解得极点为 0.5,零点为 0。接下来,引入单位圆及其 z 平面的概念来讨论极零点是如何对频率幅值响应包络产生影响的。单位圆,顾名思义,指半径为 1,圆心位于原点的圆,它在具有虚轴与实轴的二维“ z 平面”上的表示如图 4.2。传统意义上,极点在 z 平面上用符号“ $\times$ ”表示,零点用符号“ $\circ$ ”表示。

设单位圆以单位增益为半径,数字角频率 θ 以逆时针方向旋转绘制,对应扫过的频率范围 0 至 π (即 DC 分量频率到奈奎斯特限制频率),如图 4.3 所示,其中系统的输入为 $1 \cdot e^{j\theta}$ 。而频率响应的定义是系统受幅值与相位不变而频率变化的输入信号激励时,系统的幅值和相位响应。 $e^{j\theta}$ 本质上就是这种类型的输入信号——当输入信号保持幅值为 1 且初相位为 0 时,只有频率经过系统(或称传递函数),可通过频率观察系统幅值和相位响应。对于上述差分方程,极零点分布见图 4.2。如果极零点的根是 $a+jb$ 的复数形式,则极零点的位置会在直线 $b=0$ 的上侧或下侧。

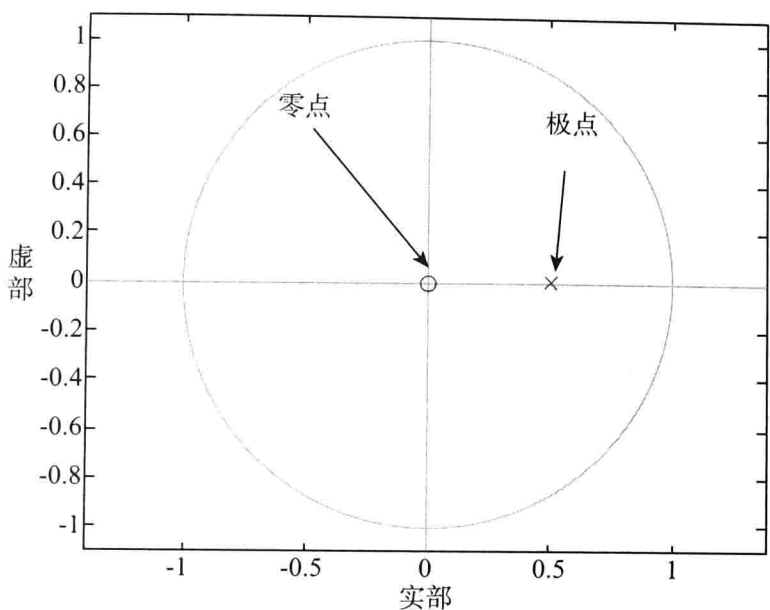


图 4.2 极零点分别位于 0.5 和 0 的单位圆

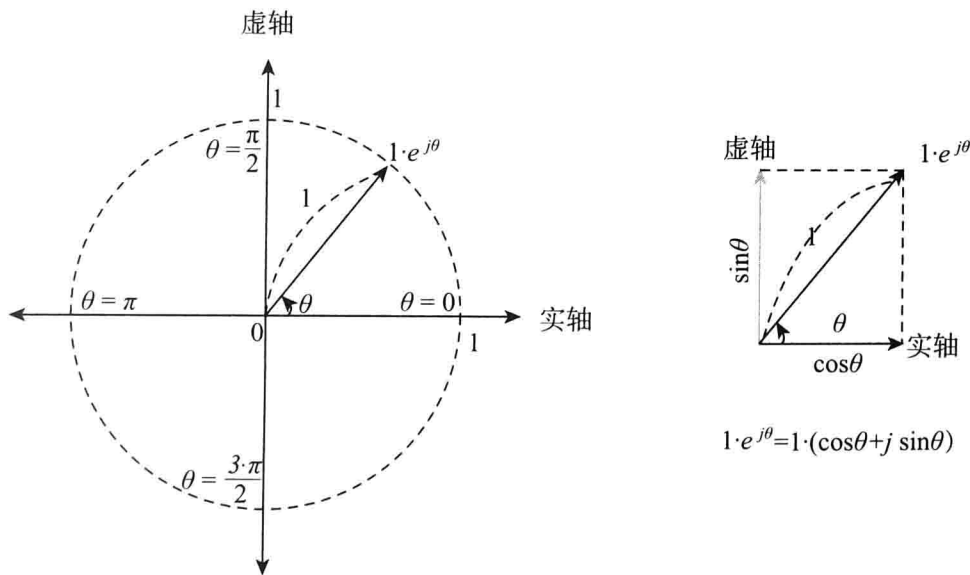


图 4.3 单位圆与欧拉公式解析

接下来,最后一步是在 z 平面上按照极零点所提供的信息绘制幅值响应草图。

对于频率响应,令人感兴趣的频率范围是从 0 至 $f_s/2$,或 0 至 π 弧度,它对应着单位圆的第一、第二象限。如前所述,极点具有共振特征(增大幅值),而零点具有阻尼特征(降低幅值),则系统的幅值响应特征如图 4.4 所示。由图可知,极点对幅值 $|H(z)|$ 的共振影响在 0 弧度处最大,而零点的阻尼影响在 π 弧度处最大——极点最接近 DC 分量,而零点最接近奈奎斯特限制频率。恰在 $\pi/2$ 弧度处,是一个极零相消点——在该频点上,由于零极点的作用相同而极性而反(即一个推一个挽),因此互相抵消。 $|H(z)|$ 的特征及其与极零点的关系也可用 4.1 节中 $|Y(z)|$ 与 $|X(z)|$ 之比表示:

$$|H(z)| = \frac{|Y(z)|}{|X(z)|}$$

(4.20)

利用本例的差分方程,有下列传递函数关系:

$$|H(z)| = \left| \frac{z}{z - 0.5} \right| = \frac{|z|}{|z - 0.5|}$$

(4.21)

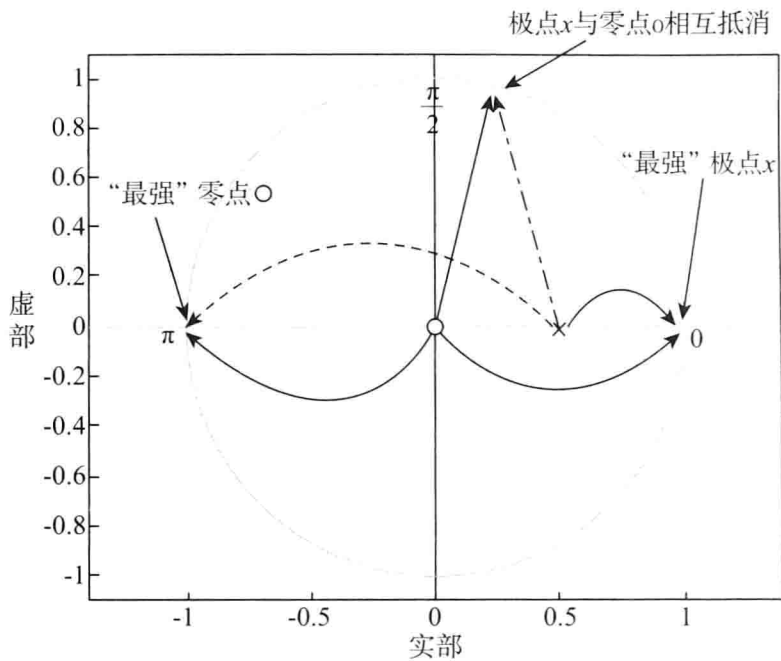


图 4.4 极零点的重要特征

由上式知,分母和分子是欧氏距离,而 $H(z)$ 是 $Y(z)$ 和 $X(z)$ 的欧氏距离之比。如图 4.5 所示,幅值 $|H(z)| = |Y(z)| / |X(z)|$ 仍在 0 弧度处最大,在 π 弧度处最小,当 $|Y(z)| = |X(z)|$ 时幅值 $|H(z)|$ 等于 1,在 $\pi/2$ 弧度前的幅值与图 4.4 的结果相同。此例中, $|Y(z)|$ 分量为常数,并位于 $z=0$ 点(单位圆的圆心),而 $|X(z)|$ 分量在 π 弧度前随 $|H(z)| (= |Y(z)| / |X(z)|)$ 的减小而增大。因此,各幅值的谷、峰及相消位置都可以在 z 平面通过观察极零点而得到,而展开后的频率响应草图如图 4.6。图 4.7 则给出了该差分方程精确的频率响应曲线。

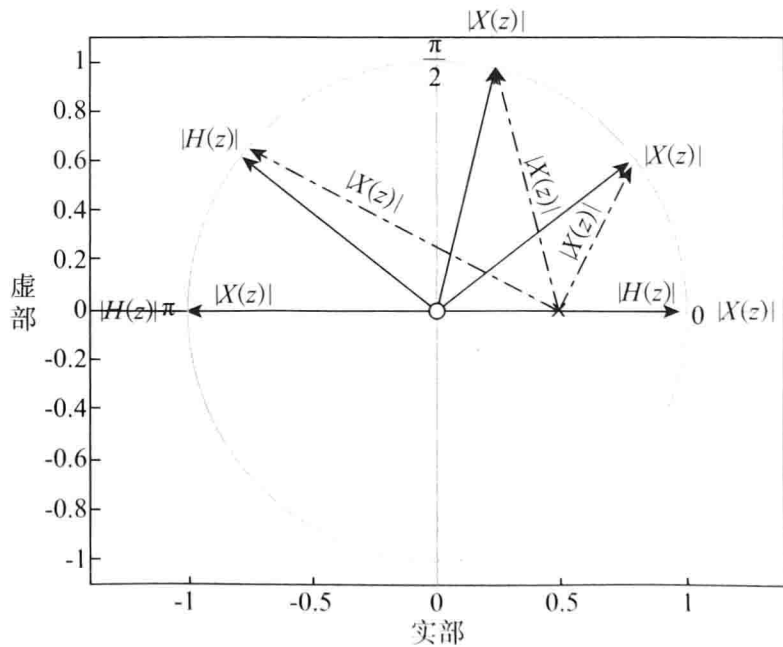


图 4.5 以从 0 至 π 的扫频绘制的全幅值

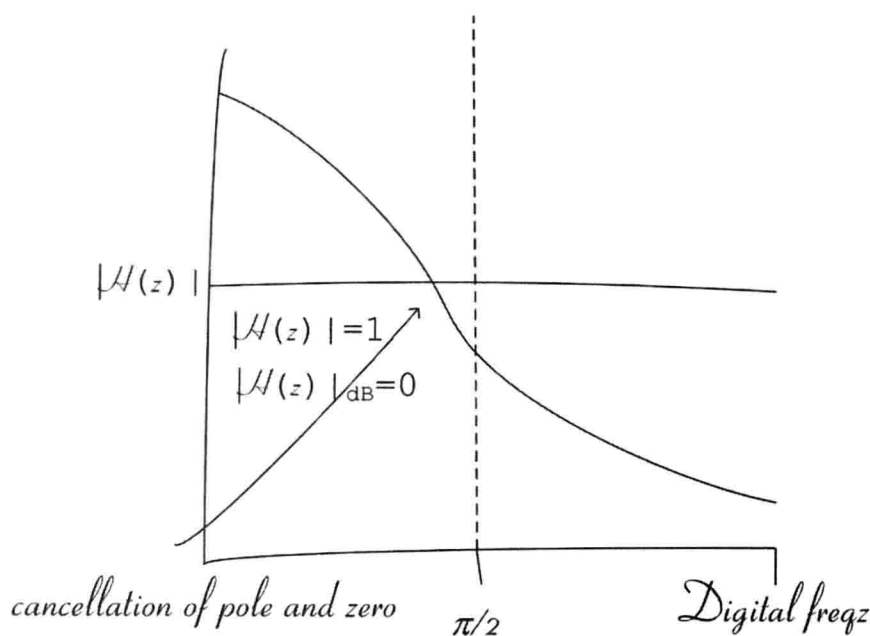


图 4.6 由极零点扩展的系统 $y[n] = x[n] + 0.5y[n-1]$ 的频率响应草图

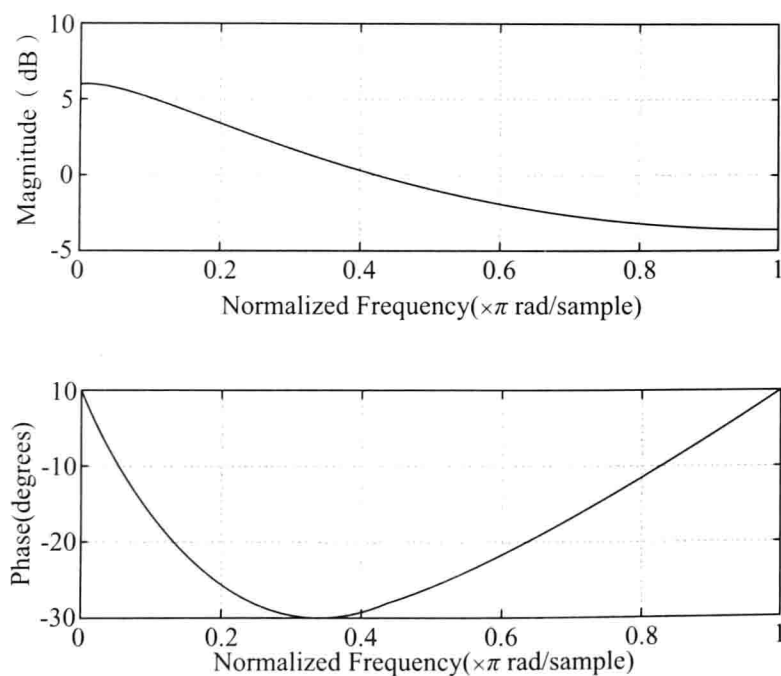


图 4.7 由计算机精确绘制的系统 $y[n] = x[n] + 0.5y[n-1]$ 的幅值频率响应与相位频率响应
(上图为幅值响应, 下图为相位响应, 对应的 MATLAB[®] 函数为 `FREQZ(1, [1 -0.5])`)

4.4 更“复杂”的系统

通常情况下, 求解 z 变换的根时会得到复数解。也就是说, 与上述简单的例子不同, 零极点往往在 z 平面上以 $a+jb$ 的复数形式分布。例如, 若一个全零系统 (FIR 系统) 的传递函数的 z 变换可用下式表示:

$$H(z) = a \cdot z^2 + b \cdot z + c \quad (4.22)$$

则该二阶多项式的通解为:

$$z = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (4.23)$$

而根号下的部分小于零时,如式 4.24,系统将有复数解,则 z 平面上的零极点位置会分散至实轴和虚轴的各个方位。

$$b^2 - 4ac < 0 \quad (4.24)$$

最重要的是,两个复数根会成为一组共轭对(conjugate pair),如式 4.25:

$$z = a \pm jb \quad (4.25)$$

则两个根关于 z 平面上的实轴 $b=0$ 对称。这些根存在于分母(极点)还是分子(零点)上,取决于系统的差分方程和滤波器的结构。

5. 收敛域 ROC

另一个关于 z 变换的重要概念称为“收敛域 ROC”(region of convergence)。ROC 的重要性在于它定义了 z 变换存在的区域。ROC 的正式定义是 z 变换和的收敛区间,并对于脉冲序列 $h[n]$ 满足式 5.1:

$$\sum_{n=-\infty}^{n=+\infty} |h[n] \cdot z^{-n}| < \infty \quad (5.1)$$

式 5.1 是一个满足有界收敛条件的几何级数,因此,可由式 5.2 展开为负部和正部:

$$\sum_{n=-\infty}^{n=+\infty} |h[n] \cdot z^{-n}| = \underbrace{\sum_{n=-\infty}^{n=-1} |h[n] \cdot z^{-n}|}_{\text{负部}} + \underbrace{\sum_{n=0}^{n=+\infty} |h[n] \cdot z^{-n}|}_{\text{正部}} \quad (5.2)$$

由上式可知,该系统可以分为非因果部分和因果部分,负部求和部分为非因果部分,正部求和部分为因果部分。为了简化该式,通常可从几何级数来考察 z 变换的收敛域。几何级数求和的一般形式如下:

$$S_N = \sum_{n=-\infty}^{n=+\infty} z^n = 1 + z^1 + \cdots + z^{N-1} + z^N \quad (5.3)$$

将和 S_N 乘以 z , 得到以下关系:

$$z \cdot S_N = z + z^2 + \cdots + z^{N-1} + z^N + z^{N+1} \quad (5.4)$$

为了消除 $z \cdot S_N$ 与 S_N 之间的常量,用式 5.4 减去式 5.3,得式 5.5 和 5.6:

$$S_N - z \cdot S_N = 1 - z^{N+1} \quad (5.5)$$

$$S_N(1 - z^{-1}) = 1 - z^{-N-1} \quad (5.6)$$

随着进一步简化,表述总和 S_N 的简洁的几何级数形式如式 5.7 所示:

$$S_N = \frac{1 - z^{N+1}}{1 - z}, \text{ 当 } z \neq 1 \text{ 时} \quad (5.7)$$

式 5.7 中 $z=1$ 时,由于“除以 0”导致 S_N 变得无穷大。 N 为 ∞ 时, S_N 可进一步简化为式 5.8,条件是 $0 < |z| < 1$ (纯小数)。若 $|z| > 1$, z^∞ 也为无穷大,会使 S_∞ 变为无穷大。而如果 $|z| < 1$, z^∞ 将趋向于 0 使 S_N 有界而变得稳定。

$$S_\infty = \frac{1 - z^{\infty+1}}{1 - z} = \frac{1 - z^\infty}{1 - z} = \frac{1}{1 - z}, \text{ 当 } |z| < 1 \text{ 时} \quad (5.8)$$

回到式 5.2 收敛域 ROC 的定义,负部求和部分可通过幂 z^{-n} 和 z^{+n} 的极性代换改写成下式,同时求和项序号 n 的特征也会发生变化。

$$\sum_{n=-\infty}^{-1} |h[n] \cdot z^{-n}| = \sum_{n=1}^{\infty} |h[-n] \cdot z^n| \quad (5.9)$$

上述来自于式 5.7 的级数与式 5.3 的形式非常相似,差别在于式 5.9 中 $h[-n]$ 的系数为 z 的幂,而式 5.3 为 0 阶次幂。为了简化该式,令 $h[-n]=1$,消除式 5.9 中 0 阶 z 分量($z^0=1$),并定义求和范围为 $n=0 \sim \infty$,可将式 5.9 写成式 5.4 的形式如下:

$$\sum_{n=-\infty}^{-1} |h[n] \cdot z^{-n}| = \sum_{n=1}^{\infty} |1 \cdot z^n| = \sum_{n=0}^{\infty} |z^n - 1| \quad (5.10)$$

如前所述,只计算从 0 至 N 阶的几何级数之和如下式 5.11:

$$\sum_{n=0}^N |z^n - 1| = \left| \frac{1 - z^{N+1}}{1 - z} - 1 \right| \quad (5.11)$$

若 $|z| < 1$,该求和将收敛如下:

$$\begin{aligned} \left| \frac{1 - z^{N+1}}{1 - z} - 1 \right|_{N=\infty} &= \left| \frac{1 - 0}{1 - z} - 1 \right| = \left| \frac{1 - (1 - z)}{1 - z} \right| \\ &= \left| \frac{z}{1 - z} \right| = \left| \frac{z}{1 - z} \cdot \frac{z^{-1}}{z^{-1}} \right| = \left| -\frac{1}{1 - z^{-1}} \right|, \text{ 当 } |z| < 1 \text{ 时} \end{aligned} \quad (5.12)$$

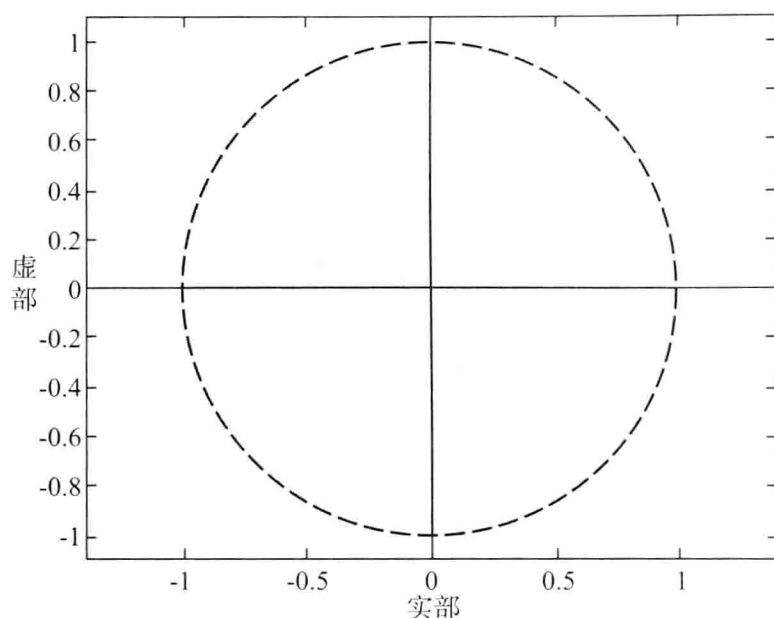
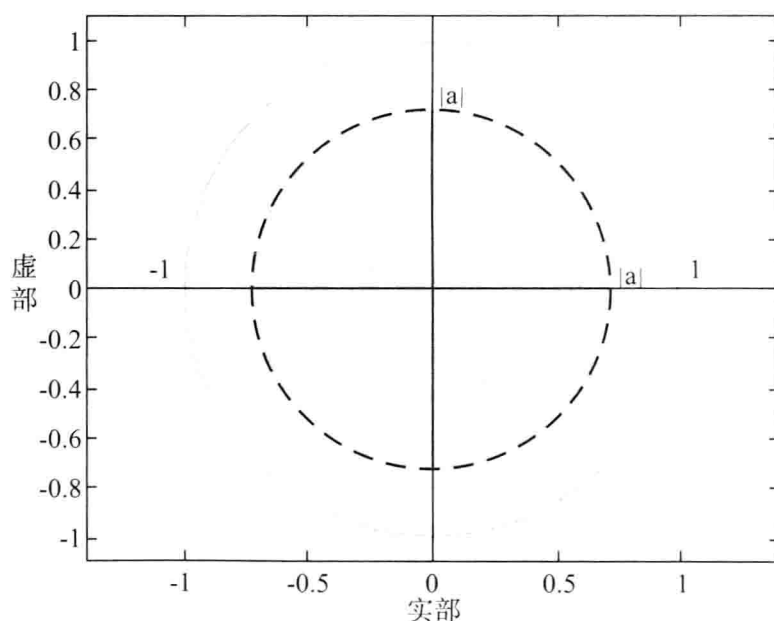
这一收敛域也是 z 变换的定义域。对于一个既定系统, $h[-n]=1$ 时,收敛域 ROC 位于如图 5.1 所示的单位圆内(不包括边界 $|z|=1$)。

再考虑一个更为有趣的非因果系统的脉冲响应序列 $h[n]$,如下式,接下来讨论它的收敛域 ROC。

$$h[n] = \begin{cases} a^n, & n < 0 \\ 0, & \text{其他} \end{cases} \quad (5.13)$$

对区域 $-N \sim -1$ 的 z 变换分量进行求和,由于这是一个非因果系统,则有式 5.14。当 $N=\infty$,有式 5.15,令 $|a^{-1}z| < 1$,收敛域图像如图 5.2 所示,其中不等式可改写为式 5.16。

$$\begin{aligned} \sum_{n=-N}^{-1} h[n] \cdot z^{-n} \Big|_{h[n]=a^n} &= \sum_{n=-N}^{-1} a^n \cdot z^{-n} = \sum_{n=0}^N a^{-n} \cdot z^n - 1 \\ &= \sum_{n=0}^N (a^{-1} \cdot z)^n - 1 = \frac{1 - (a^{-1}z)^{N+1}}{1 - a^{-1}z} - 1 \end{aligned} \quad (5.14)$$

图 5.1 $h[-n] = 1, |z| < 1$ 时的收敛域 ROC图 5.2 非因果系统 $h[n] = a^n, |z| < |a|$ 的收敛域 ROC

$$\left. \frac{1 - (a^{-1}z)^{N+1}}{1 - a^{-1}z} - 1 \right|_{n=\infty} = -\frac{1-0}{1-a^{-1}z} - 1 \quad (5.15)$$

$$= \frac{a^{-1}z}{1 - a^{-1}z} = -\frac{1}{1 - a^{-1}z}, |a^{-1}z| < 1$$

$$|a^{-1} \cdot z| < 1 \rightarrow |z| < |a| \quad (5.16)$$

以上推导过程表明, $h[n] = a^n$ 的 z 变换收敛至式 5.15, 即零点位于 $z=0$ 处, 又由式 5.15 知 $1/(1-a^{-1}z) = z/(z-a^{-1})$, 则极点位于 $z=a$ 处。由于该 z 变换不收敛于极点, 则 z 变换的极点不会出现在单位圆上, 收敛域本身以极点为边界。由图 5.2 可知, 零点位于原点, 极点如前所述位于 $z=a$ 处。

此外还需要注意, $h[n]$ 也可用单位阶跃函数表示, 即:

$$h[n] = a^n \cdot u[-n-1] \quad (5.17)$$

由上式可知, 指数项只有在负 n 次幂时存在。也就是说, 因为 $u[-n-1]$ 只有在 n 为负整数时存在(由非因果系统的定义可知), 所以当 $n \geq 1$ 时, a^n 可改写成 a^{-n} 。

5.1 因果系统的收敛域

因果系统本质上相当于式 5.2 的正部求和部分, 对于 $h[n]$, 可用与非因果系统相同的方法得出其收敛域 ROC, 如下:

$$h[n] = \begin{cases} a^n, & n \geq 0 \\ 0, & \text{其他} \end{cases} \quad (5.18)$$

$$\begin{aligned} \sum_{n=-\infty}^{n=+\infty} |h[n] \cdot z^{-n}| &= \sum_{n=0}^{n=\infty} |h[n] \cdot z^{-n}| \\ &= \sum_{n=0}^{n=\infty} |a^n \cdot z^{-n}| = \sum_{n=0}^{n=\infty} |(a \cdot z^{-1})^n| \end{aligned} \quad (5.19)$$

观察式 5.19 可知, 求和总项为 N 时得:

$$\sum_{n=0}^{n=N} (a \cdot z^{-1})^n = \frac{1 - (a \cdot z^{-1})^{N+1}}{1 - a \cdot z^{-1}} \quad (5.20)$$

当 N 为无穷大时, 有如下结果:

$$\sum_{n=0}^{n=\infty} (a \cdot z^{-1})^n = \frac{1 - (a \cdot z^{-1})^{\infty+1}}{1 - a \cdot z^{-1}} = \frac{1}{1 - a \cdot z^{-1}} \text{ for } |a \cdot z^{-1}| < 1 \quad (5.21)$$

式 5.21 中的不等式可改写为式 5.22, 如图 5.3 所示。

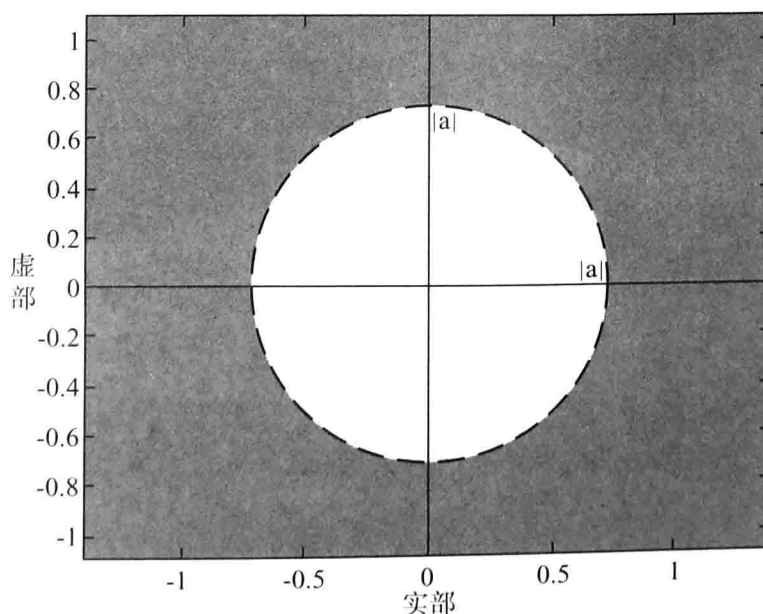


图 5.3 因果系统 $h[n] = a^n, |a| < |z|$ 的收敛域 ROC

$$|a \cdot z^{-1}| = \left| \frac{a}{z} \right| \text{ 或 } |a| < |z| \tag{5.22}$$

因此,因果系统的收敛域 ROC 位于单位圆外。

5.2 复合因果系统

对于同时包含因果和非因果部分的系统,如式 5.23,其收敛域 ROC 为图 5.4 所示的一个二维圆环。反映其收敛域 ROC 的不等式如式 5.24 和 5.25。

$$h[n] = a^n \cdot u[n] + b^n \cdot u[-n - 1] \tag{5.23}$$

由上述分析可知,因果分量的收敛域如式 5.24,非因果分量的收敛域如式 5.25。

$$|z| > |a| \tag{5.24}$$

$$|z| < |b| \tag{5.25}$$

因为 LTI 系统的线性特征,可将以上两个部分整合为一个不等式 5.26,则复合因果系统的收敛域如图 5.4 和式 5.26:

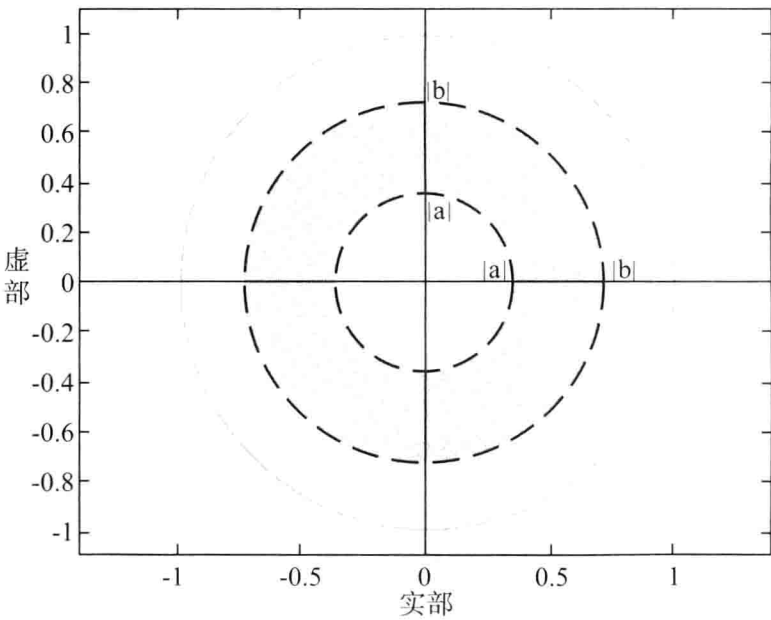


图 5.4 复合因果系统 $|a| < |z| < |b|$ 的收敛域

$$|a| < |z| < |b| \tag{5.26}$$

5.3 收敛域 ROC 小结

由之前几个小节对收敛域 ROC 的讨论可知,极点不是位于收敛域内就是位于所定义的收敛域的边界上。第 5 节所讨论的因果系统、非因果系统与复合因果系统的收敛域总结如表 5.1,其中 a 为极点的半径。

表 5.1 各因果类型系统的收敛域 ROC

系统类型	系统与 ROC 特征
因果	$ a < z $ (图 5.3) 所有极点都在单位圆内(对其系统稳定性的讨论见下节)
非因果	$ z < a $ (图 5.2)
复合因果	$ a < z < b $ (图 5.4)

6. 稳定性与单位圆

简而言之,系统的稳定性取决于第 5 章 4.2 节中所介绍的 z 平面上极点的位置。FIR 系统或没有极点的系统由于没有反馈分量被定义为稳定的——其实没有任何一个系统一开始就是不稳定的,当然除非系统的输入会随时间而增大。稳定性的正式定义如式 6.1,它在本质上是指 在某个时刻,系统的脉冲响应最终会衰减至 0(系统有界),则该系统是稳定的。

$$\sum_{n=-\infty}^{n=+\infty} |h[n]| < \infty$$

(6.1)

从 z 平面和极点的位置来看,很容易解释有极点的系统会包含反馈分量。如第 5 章所述, FIR 系统的脉冲响应必然是有限的,当输入停止时,系统最终也会停止。IIR 系统则相反,即使输入停止,从技术的角度看,系统也不会停止。而由于数字系统的计算精度有限,若一个 IIR 系统是稳定的,在某一时刻后它的输出也将为 0(设数字系统有足够的比特数来表示一个纯小数,如 10^{-53} ,但在实际情况下该数字都等于 0)。既然 FIR 系统本质上是稳定的,因此不必担心它的稳定性。对于 IIR 系统,则必须检验其稳定性(它是极点特征的函数)。

IIR 系统稳定的充分条件是,系统的所有极点都在单位圆内。极点特征可由第 5 章 4.2 节所介绍的差分方程及其脉冲响应表示如下:

$$y[n] = x[n] + a \cdot y[n - 1]$$

(6.2)

$$h[k] = \delta[k] + a \cdot h[k - 1] = a^k$$

(6.3)

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 - az^{-1}}$$

(6.4)

为了使系统稳定,必须满足式 6.5 的无限求和条件。即在某个时刻,系统的脉冲响应(如一个快速开/关信号)会衰减至 0,脉冲响应之和必须是有界的。

$$\sum_{n=0}^{n=+\infty} |h[n]| < \infty$$

(6.5)

再用几何级数进行展开,则对于该 IIR 系统,上述求和结果如式 6.6:

$$\sum_{n=0}^{n=+\infty} |h[n]| = \sum_{n=0}^{n=N} |a^n| \Big|_{N=\infty} = \frac{1 - |a|^{N+1}}{1 - |a|} \Big|_{N=\infty}$$

(6.6)

$$|a| < 1$$

(6.7)

要使式 6.6 收敛,因存在 $|a|^{N+1}$ 分量, $|a|$ 必须小于 1;若 $|a| > 1$,式 6.6 将无界。由 z 平面和第 5 节所述可知,极点必须在单位圆内,则系统的收敛域 ROC 可由式 6.8 求得:

$$\sum_{n=-\infty}^{n=+\infty} |h[n] z^{-n}| < \infty \tag{6.8}$$

因此,对于因果系统,极点在单位圆内且收敛域 ROC 包含单位圆,则系统稳定。对于非因果系统极点的情况正好相反——所有极点必须在单位圆外且同上所述收敛域 ROC 包含单位圆。这是因为在非因果系统中当极点位于单位圆外时,脉冲响应 $h[n]$ 增大至负次幂时的强度与第 5 节所讨论的极点特征条件相反。

7. 逆 Z 变换

简而言之,逆 z 变换是从“ z 域”(z-domain)到时域的变换。对于给定的式 7.1, $H(z)$ 的逆 z 变换将得到该系统的脉冲响应 $h[n]$,如式 7.2:

$$H(z) = \frac{Y(z)}{X(z)} \tag{7.1}$$

$$Z^{-1}[H(z)] = h[n] \tag{7.2}$$

同样以上例差分方程为 $y[n] = x[n] + y[n-1]$ 的 IIR 系统为例,其逆 z 变换如式 7.3:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 - az^{-1}} \tag{7.3}$$

如果出于某种原因想回到时域差分方程,则将 $H(z)$ 乘以 2 得:

$$\frac{Y(z)}{X(z)} = \frac{1 \cdot 2}{1 - az^{-1}} \tag{7.4}$$

$$Y(z)(1 - az^{-1}) = 2 \cdot X(z) \tag{7.5}$$

$$Y(z) = 2 \cdot X(z) + az^{-1} \cdot Y(z) \tag{7.6}$$

最终通过第 4 节的变量代换,即 $x[n] \rightarrow X(z)$, $y[n] \rightarrow Y(z)$,延迟分量由 $g[n-L] \rightarrow z^{-1}G(z)$,可得到该系统的时域差分方程如式 7.7:

$$y[n] = 2 \cdot x[n] + a \cdot y[n - 1] \tag{7.7}$$

逆 z 变换的强大不仅因为可得到系统的差分方程,更重要的是它可以直接通过 $Y(z)$ 计算输出序列 $y[n]$,或通过 $H(z)$ 计算系统的脉冲响应序列 $h[n]$ 。获得系统脉冲响应序列的常见方法包括长除法(long division method),泰勒级数展开法(Taylor series method)和留数定理法(residue theorem method)。这些方法将在后续几个小节中简要地加以介绍。

7.1 长除法

回到式 4.2,信号 $x[n]$ 的 z 变换定义如式 7.8。序列 $x[n]$ 当然可以是任何信号,包括脉冲序列 $h[n]$ 。如果将 z 变换 $X(z)$ 表示为多项式,如式 7.9,可发现一个联系 $X(z)$ 与 $x[n]$ 的属

性——即输入序列 $x[n]$ 是 $X(z)$ 中 z^{-n} 分量的系数(乘积因子或权重)。

$$X(z) = \sum_{n=-\infty}^{+\infty} x[n] z^{-n} \quad (7.8)$$

$$= \cdots + x[-1] \cdot z^1 + x[0] \cdot z^0 + x[1] \cdot z^{-1} + \cdots$$

$$X(z) = \cdots + a \cdot z + b + c \cdot z^{-1} + d \cdot z^{-2} + e \cdot z^{-3} + \cdots \quad (7.9)$$

通过比较式 7.8 和 7.9, 有如下关系:

$$\begin{aligned} & \cdots \\ x[-1] &= a \\ x[0] &= b \\ x[1] &= c \\ x[2] &= d \\ x[3] &= e \\ & \cdots \end{aligned}$$

因此, 已知输入信号的变换 $X(z)$ 或任何 z 变换过程的多项式形式, 可求解输入序列 $x[n]$, 或已知系统输出信号的 z 变换 $Y(z)$ 和传递函数的 z 变换 $H(z)$, 可求解系统的输出 $y[n]$ 和脉冲响应 $h[n]$ 。上例的 $X(z)$ 非常简单, 没有反馈分量——即没有分母。但在一般信号处理中, 经常会碰到 IIR 系统的差分方程, 使求解由 z^{-n} 的幂表示的多项式序列的过程变得更为复杂。而这一过程就适合用长除法进行。顾名思义, 长除法就相当于小学所学过的除法。

例如, 设传递函数 $H(z)$ 如式 7.10, 包含两个极点和两个零点:

$$H(z) = \frac{z^2 + z}{(z + 0.5)(z - 0.3)}, |z| > 0.5 \quad (7.10)$$

要进行长除法, 首先将 $H(z)$ 展开如式 7.11:

$$H(z) = \frac{z^2 + z}{(z + 0.5)(z - 0.3)} = \frac{z^2 + z}{z^2 - 0.3z + 0.5z - 1.5} = \frac{z^2 + z}{z^2 + 0.2z - 1.5} \quad (7.11)$$

然后运用长除法可得式 7.12:

$$\begin{array}{r} 1 - 0.8z^{-1} + 0.8z^{-2} \cdots \\ z^2 + z - 1.5 \overline{) z^2 + 0.2z - 1.5} \\ \underline{z^2 + z + 1.5} \\ 0 - 0.8z + 0 \\ - 0.8z - 0.8 + 1.5z^{-1} \\ \underline{0 + 0.8 - 1.5z^{-1}} \\ 0.8 + 0.8z^{-1} - 1.2z^{-2} \\ \cdots \end{array} \quad (7.12)$$

由传递函数的 z 变换 $H(z)$ 的定义可知, z^{-n} 的系数与系统脉冲响应序列 $h[n]$ 的各分量相对应, 则如前所述, 对于该因果系统, 可直接得到序列 $h[n]$ 的值:

$$\begin{aligned}
 h[0] &= 1 \\
 h[1] &= 0.8 \\
 h[2] &= 1.32 \\
 &\dots
 \end{aligned}$$

7.2 泰勒级数展开法

第5节已经使用了泰勒级数展开来求解收敛域 ROC, 并且注意到 $|z| < 1$ 收敛时, 该无限求和形式如式 7.12。

$$S_N = \sum_{n=0}^{n=N} z^n = 1 + z^1 + \dots + z^{N-1} + z^N \quad (7.13)$$

$$\sum_{n=0}^{n=N} z^n = \frac{1}{1-z} \quad (7.14)$$

之前有该求和形式与泰勒级数展开式的一般关系如下:

$$\sum_{n=0}^{n=\infty} (a \cdot z^{-1})^n = \frac{1 - (a \cdot z^{-1})^{\infty+1}}{1 - a \cdot z^{-1}} = \frac{1}{1 - a \cdot z^{-1}}, \text{ 当 } |a \cdot z^{-1}| < 1 \text{ 时} \quad (7.15)$$

从逆 z 变换的角度来看, 式 7.15 将产生一个形式为 $h[n] = a^n$ 的时间序列 (见 5.1 节)。因此, 如果将 z 变换写成类似式 7.15 的形式, 可计算出时间序列 $h[n]$, 这就是该特殊方法在本质上所解决的问题。以式 7.16 给定的传递函数的 z 变换 $H(z)$ 为例, 目标是将 $H(z)$ 写成类泰勒级数展开的形式。

$$H(z) = \frac{1 - a + z - bz^{-2}}{1 - bz^{-2} - az^{-1} + abz^{-3}}, |z| > a, a > b \quad (7.16)$$

对上式进行因数分解有式 7.17。这是一个线性系统, 因此可将一个复杂问题分解为多个易于处理的小问题——注意到分解结果的两个分量都具有泰勒级数结构。

$$\begin{aligned}
 H(z) &= \frac{1 + z^{-1} - (a+b)z^{-2}}{1 - bz^{-2} - az^{-1} + abz^{-3}} = \frac{1 + z^{-1} - bz^{-2} - az^{-2}}{(1 - az^{-1})(1 - bz^{-2})} \\
 &= \frac{1}{1 - az^{-1}} + z^{-1} \cdot \frac{1}{1 - bz^{-2}}
 \end{aligned} \quad (7.17)$$

因此, 将式 7.17 改写为 7.18, 并展开为式 7.19:

$$H(z) = \sum_{n=0}^{n=\infty} (a \cdot z^{-1})^n + z^{-1} \cdot \sum_{n=0}^{n=\infty} (b \cdot z^{-2})^n \quad (7.18)$$

$$\begin{aligned}
 H(z) &= (a \cdot z^{-1})^0 + (a \cdot z^{-1})^1 + (a \cdot z^{-1})^2 + \dots + z^{-1} \\
 &\quad \times [(b \cdot z^{-2})^0 + (b \cdot z^{-2})^1 + (b \cdot z^{-2})^2 + \dots] \\
 &= 1 + az^{-1} + a^2z^{-2} + a^3z^{-3} + \dots + z^{-1} + bz^{-3} + b^2z^{-5} \\
 &\quad + b^3z^{-7} + \dots \\
 &= 1 + (a+1)z^{-1} + (a^2)z^{-2} + (a^3+b^1)z^{-3} \\
 &\quad + (a^4)z^{-4} + (a^5+b^2)z^{-5} + \dots
 \end{aligned} \quad (7.19)$$

所以,如果 $H(z)$ 可表示为式 7.15 的形式,则对延迟分量 z^{-k} 合并同类项得到延迟系数 $h[n]$ 后,通过逆 z 变换得如式 7.20 所示的序列 $h[n]$ 。即式 7.19 的最后一行用延迟量 z^{-k} 及其各自的系数或权重表示。

$$h[n] = \begin{cases} 1, & n = 0 \\ a + 1, & n = 1 \\ a^2, & n = 2 \\ a^n + b^{(2n+1)-2}, & n = \text{奇数}, n > 2 \\ a^n, & n = \text{偶数}, n > 2 \\ 0, & \text{其他} \end{cases} \quad (7.20)$$

7.3 曲线积分与留数定理

本章将要介绍的最后一种计算逆 z 变换的方法用到了一些复变函数理论 (complex variable theory) 中的概念。由复变函数理论可知,系统的脉冲响应 $h[n]$ 及其 z 变换 $H(z)$ 有如式 7.21 的关系。其中,积分为在收敛域 ROC 内以原点 ($z=0$) 为起始位置的逆时针方向的封闭曲线积分。

$$x[n] = \frac{1}{2\pi} \oint X(z) \cdot z^{n-1} dz \quad (7.21)$$

直接计算式 7.21 的时间序列往往非常困难,但由于大多数 DSP 系统的 z 变换都可表示为有理函数 (如两个多项式之比 $Y(z)/X(z)$), 因此可运用更简单的柯西留数定理 (Cauchy's residue theorem) 求解 $x[n]$ 。由柯西留数定理知,式 7.21 的左边可表示为式 7.22, 其中 p_k 指 K 个极点 $p_1, p_2, \dots, p_k, \dots, p_K$ 中的第 k 个极点,符号“Res”表示极点 p_k 的留数。

$$\frac{1}{2\pi j} \oint H(z) \cdot z^{n-1} dz = \sum_{k=1}^K \text{Res}_{z \rightarrow p_k} [H(z) \cdot z^{n-1}] \quad (7.22)$$

可用式 7.23 获得留数的计算结果,其中 p_k 为第 m 阶极点,则式 7.24 用来计算第一阶极点 ($m=1$ 时,式 7.23 可简化为式 7.24)。

$$\text{Res}_{z=p_k} [H(z) \cdot z^{n-1}] = \frac{1}{(m-1)!} \cdot \lim_{z \rightarrow p_k} \frac{d^{m-1}}{dz^{m-1}} [(z-p_k) \cdot H(z) \cdot z^{n-1}] \quad (7.23)$$

$$\text{Res}_{z=p_k} [H(z) \cdot z^{n-1}] = \lim_{z \rightarrow p_k} (z-p_k) \cdot H(z) \cdot z^{n-1} \quad (7.24)$$

式 7.22 实际上看起来比下述计算式 7.25 的脉冲响应序列 $h[n]$ 的例子更复杂。

$$H(z) = \frac{z(z+1)}{(z-0.5)(z+0.3)}, |z| > 0.5 \quad (7.25)$$

由式 7.25 可找到该系统的两个极点 (即分母的根)。

$$z - 0.5 = 0 \rightarrow z = 0.5 \quad (7.26)$$

$$z + 0.3 = 0 \rightarrow z = -0.3 \quad (7.27)$$

两个极点都是的一阶极点 ($m=1$), 因此 $h[n]$ 由式 7.24 计算如下:

$$\begin{aligned}
h[n] &= \sum_{k=1}^2 \operatorname{Res}_{z=p_k} [H(z) \cdot z^{n-1}] \\
&= \operatorname{Res}_{z=0.5} [H(z) \cdot z^{n-1}] + \operatorname{Res}_{z=-0.3} [H(z) \cdot z^{n-1}] \\
&= (z-0.5) \cdot \frac{z(z+1)}{(z-0.5)(z+0.3)} z^{n-1} \Big|_{z=0.5} \\
&\quad + (z+0.3) \cdot \frac{z(z+1)}{(z-0.5)(z+0.3)} z^{n-1} \Big|_{z=-0.3} \quad (7.28) \\
&= \frac{(z+1)}{(z+0.3)} \cdot z^n \Big|_{z=0.5} + \frac{(z+1)}{(z-0.5)} \cdot z^n \Big|_{z=-0.3} \\
&= \frac{1.5}{0.2} 0.5^n - \frac{0.7}{0.8} (-0.3)^n \\
&= 7.5 \cdot (0.5)^n - 0.875 \cdot (-0.3)^n
\end{aligned}$$

对于更复杂的系统和高阶极点,虽然计算过程更为繁琐,但实际机制与上述过程相似。

为了更好地理解曲线积分和留数定理,本节的末尾将证明式 7.22。首先将式 4.2X(z) 的定义式代入式 7.22,有:

$$\begin{aligned}
x[n] &= \frac{1}{2\pi j} \oint X(z) \cdot z^{n-1} dz \\
&= \frac{1}{2\pi j} \oint \left\{ \sum_{m=-\infty}^{+\infty} x[m] z^{-m} \right\} \cdot z^{n-1} dz \quad (7.29) \\
&= \sum_{m=-\infty}^{+\infty} \left(x[m] \cdot \frac{1}{2\pi j} \oint z^{-m} \cdot z^{n-1} dz \right) \\
&= \sum_{m=-\infty}^{+\infty} \left(x[m] \cdot \frac{1}{2\pi j} \oint z^{n-m-1} dz \right)
\end{aligned}$$

为了证明式 7.22,用复变函数理论求解式 7.29 的积分项(Porat 1997),得:

$$\frac{1}{2\pi j} \oint z^k dz = \delta[k+1] = \begin{cases} 1, & k = -1 \\ 0, & k \neq -1, k \in \text{整数} \end{cases} \quad (7.30)$$

对于时间离散序列, $k = n - m - 1$ (见式 7.29 末尾的指数项), 将式 7.30 中的 k 用 $n - m - 1$ 代入,有:

$$\frac{1}{2\pi j} \oint z^k dz = \delta[k+1] = \frac{1}{2\pi j} \oint z^{n-m-1} dz = \delta[n-m-1+1] = \delta[n-m] \quad (7.31)$$

将上式代回式 7.29 消除积分项后,即剩下所期望得到的输入序列 $x[n]$ 。需要记住的是, δ 函数只在 $n-m=0$ 时存在,即 $m=n$ 时, δ 函数本身的幅值才为 1。

$$\sum_{m=-\infty}^{+\infty} \left(x[m] \cdot \frac{1}{2\pi j} \oint z^{n-m-1} dz \right) = \sum_{m=-\infty}^{+\infty} (x[m] \cdot \delta[n-m]) = x[n] \quad (7.32)$$

如本节所述,计算逆 z 变换的方法有很多,每个方法都有自身的优点和缺点。下节以一些与上述算法有关的 MATLAB[®] 函数对本章做一个技术性的总结。

8. 有用的 MATLAB® 工具包

MATLAB® 软件中有许多有用的函数可用来进行频率响应的计算,包括函数 $FREQZ(B, A, N)$ 和函数 $INVREQZ(H, W, NB, NA)$ 。函数 $FREQZ(B, A, N)$ 用来计算一个给定的 N 阶差分方程的幅值与相位响应, $INVREQZ(H, W, NB, NA)$ 则计算频率响应为 H 、频点为 W 以及分子与分母系数分别为 NB 和 NA 的差分方程系数。其他实用的函数还包括用来显示 z 平面及其零极点分布图的函数 $ZPLANE(Z, P)$, Z 和 P 分别为通过函数 $ROOTS(C)$ 计算得到的零极点。此外还有如函数 $GRPDELAY(B, A, N)$ 和 $IMPZ(B, A, N)$, 可用于计算系统的群延迟和脉冲响应。

9. 乐例分析

尽管下一章将关注的滤波器与 z 变换有很大关系,但 z 变换本身在音乐中应用的例子并不常见。令人感兴趣的话题是本章所讨论的稳定性问题。如上述所学到的,对于数字信号处理本身,稳定的系统非常重要。而数字信号处理又作为不同层面上音乐创作和演出灵感的源泉,系统的稳定性同样也是一个重要问题。上述章节简要介绍了在扩声系统的扬声器和传声器配置不当时经常发生的声反馈,也称为“拉森效应”(Larsen effect)。在大多数情况下,声反馈是不需要的,但这并不妨碍它被艺术家当做一种创造性的手法,成为加工音乐素材的大型工具库。流行音乐家,如 The Who 乐队的电吉他大师吉米·亨德里克斯(Jimi Hendrix)和皮特·汤森(Pete Townsend),在他们的作品《机关枪》(*Machine Gun*)中已经使用了由电吉他阵列技术产生的,包括声反馈在内的大量非稳定扩声效果。The Who 乐队像许多摇滚乐队一样,也经常在音乐作品中利用声反馈,一张值得注意的专辑《汤米》(*Tommy*)发行于 1969 年,这是他们第一张所谓的摇滚歌剧专辑,四年后于 1973 年,他们又发行了另一张专辑《四声道》(*Quadrophonia*)。电吉他的声反馈方式很有趣,因为它没有传声器—扬声器循环,而是仅靠电吉他及其拾音器和吉他功放来完成的。人们时常会看到吉他手在功率放大器前附下身去,好像在对某些未知的神灵进行一种古怪的祭拜仪式。其实这种非常有趣的姿势是很实用的——它能听取由电吉他与功率放大器产生的输入—输出闭环声反馈。功放端的增益足够大时,该闭环会形成反馈机制——即吉他弦的振动也由功率放大器或扬声器以较高增益放大,随后,功放端的高增益和吉他与功率放大器的相互接近,引发吉他弦的物理振动被再次放大而形成反馈。实际上好像是经扬声器或功率放大器放大的声音在弹奏吉他,而不是吉他手。

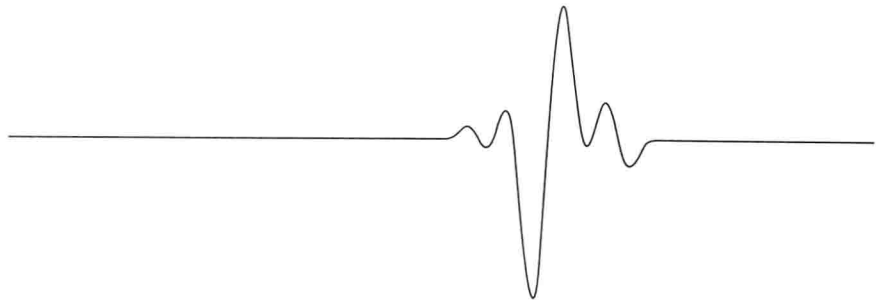
对于歌剧,另一个在作品中使用声反馈的艺术家是罗伯特·阿什利(Robert Ashely),该作曲家兼演奏家以在电视歌剧(opera-for-television)方面的成就而著名。在他于 20 世纪 60 年代初为现场人声、磁带和人声反馈而创作的作品《狼人》(*Wolfman*)中,将传声器的增益设得非常高,以造成扩声系统的轻微不稳定,有利于制造“专属”声音和声反馈效果。这首作品在音色的控制上相当出色,据说人为制造出了一种类似“沧桑而慵懒的酒吧歌手”的音色。所以并不令人奇怪的是,一些本地人认为这种声反馈啸叫让阿什利作品的标题《狼人》变得更加真实。根据后来对该歌曲的介绍(Alga Marghen 2002),阿什利的一位歌手朋友希望由莫顿·费尔德曼(Morton Feldman)来写这首歌曲,但最终没有实现。在创作《狼人》这首作品时,阿什利似乎已经知道歌手自己也不会到现场演唱这首作品,所以最后在 1964 年纽约的夏洛特·穆尔曼(Charlotte

Moorman) 音乐节上,他自己演唱了这首作品。更有趣的是,歌曲《狼人》的高分贝音量与费尔德曼的音乐风格大相径庭,不过的确,阿什利通过高增益的传声器完成了轻声细语的演唱。

参考文献与扩展阅读

- [1] Alga M. 2002. Wolfman-Robert Ashley. Alga Marghen label, Italy.
- [2] Blauert, J., Laws, P 1978. “Group Delay Distortions in Electroacoustical Systems”, Journal of the Acoustical Society of America, Volume 63(5), 1478 – 1483.
- [3] Claerbout, J. 1998. http://sepwww.stanford.edu/sep/prof/toc_html/.
- [4] Greenberg, M. D. 1988. Advanced Engineering Mathematics, Prentice Hall International Editions ISBN 0 – 13 – 011552 – 5.
- [5] Porat, B. 1997. A Course in Digital Signal Processing. Wiley and Sons.
- [6] Smith, J. O. III 2007. Introduction to Digital Filters with Audio Applications, W3K Publishing, <http://books.w3k.org/>, 2007, ISBN 978 – 0 – 9745607 – 1 – 7.
- [7] Strum, R. D. Kirk 1988. First Principles of Discrete Systems and Digital Signal Processing, Reading MA: Addiso-Wesley

第 7 章 滤 波 器



1. 引 言

过滤器在日常生活中随处可见,如水净化装置上的水过滤网,用来过滤蚊子和小飞虫的纱窗,酒吧按年龄(或其他法定标准)来过滤顾客,又或者是空调机上那些经常因为偷懒而没有定期更换的空气过滤片。在数字信号处理中,数字滤波器也会普遍存在,它们用于去除工程信号、音乐和音频应用软件中过剩的能量,这就是本章所关注的内容。严格来说,至少在信号处理中,滤波器改变信号的振幅和相位。如一个振幅为 A , 频率为 f , 初相位为 ϕ 的正弦或余弦信号,滤波时只会修改 A 和 ϕ 。接下来的几节将介绍许多种重要的滤波器,也会分享一些在声音与音乐信号处理领域中有趣的应用实例。

2. 低通、高通、带通和带阻滤波器

一种最常用也最重要的滤波器是低通滤波器(low-pass filter)。第 1 章在介绍采样与频率混叠现象时,第 2 章在讨论减采样与过采样时都简要地提到过这种滤波器。读者应该记得,在采样和模数转换中,当模拟信号的最高频率分量接近如式 2.1 的奈奎斯特限制频率时,会产生频率混叠。

$$f_{\max} < \frac{f_s}{2} \tag{2.1}$$

不满足以上不等式的情况下,一定会发生频率混叠。但是如果采样器的输入信号未知,为了不发生频率混叠,则在数字化过程中模拟信号需要一个滤波器,以去除其大于 $f_s/2$ 的频率分量。也就是说,为了避免频率混叠,必须保留低于 $f_s/2$ 的频率分量。低通滤波器就可以完成此项工作——让低频分量通过而阻止更高的频率分量,如图 2.1 所示。那些可通过并且在理想情况下不受滤波器干扰的频率域称为通带(passband)频域,而被过滤或阻止的频率域称为阻带(stopband)。图 2.1 也给出了区分通带和阻带特征的截止频率 f_c (cutoff frequency)——理想的低通滤波器会去除所有高于 f_c 的频率分量。对于防混叠滤波器(anti-filter)有 $f_c = f_s/2$ 。

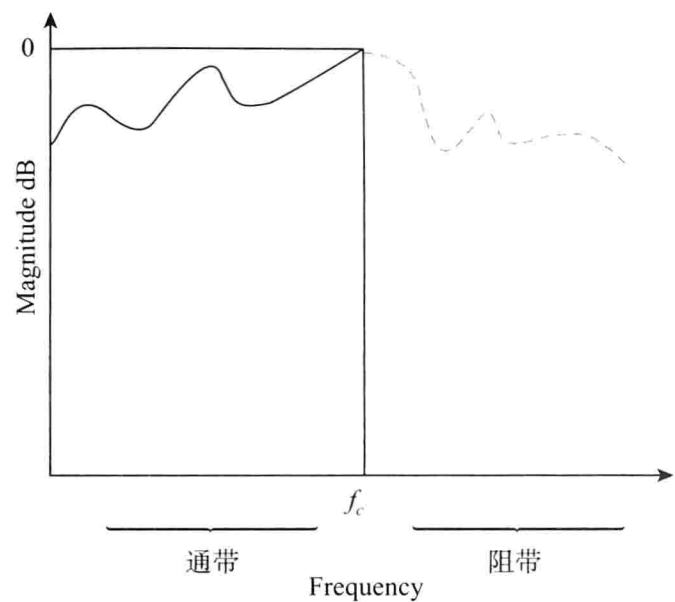


图 2.1 理想低通滤波器

图 2.2 给出了一个理想的高通滤波器 (high-pass filter), 它会去除所有低于 f_c 的频率分量。图 2.3 是一个理想的带通滤波器 (band-pass filter), 中心频率 (center frequency) f_0 以及定义通带范围的上下截止频率 (upper / lower frequency) f_1, f_2 。下一节介绍 Q 因子 (这一因子表征了带通滤波器的带通区域的锐度) 时, 将进一步阐述带通滤波器。图 2.4 则是一个理想带阻滤波器 [band-stop filter, 也称为带止滤波器 (band-reject filter) 或陷波滤波器 (notch filter)], 它与带通滤波器相反, 将阻止某一频域而保留其他频率分量。需要注意的是, 图 2.1 ~ 2.4 给出的都是理想滤波器 (idea filter), 有时也称为砖墙式滤波器 (brick-wall filter)。实际使用的滤波器很难达到砖墙式滤波器的性能, 需要许多优化参数来对其进行调整, 以满足性能需求。而截止频率 f_c 指幅值下降至通带幅值的 0.707 倍 (-3dB) 时对应的频点。系数 0.707 来自于电学上功率值的定义, 其数值为通带功率的一半。因为功率是电压的平方, 所以当功率降低一半时, 则对应幅值或电压下降至原值的 $1/\sqrt{2}$, 即 0.707。

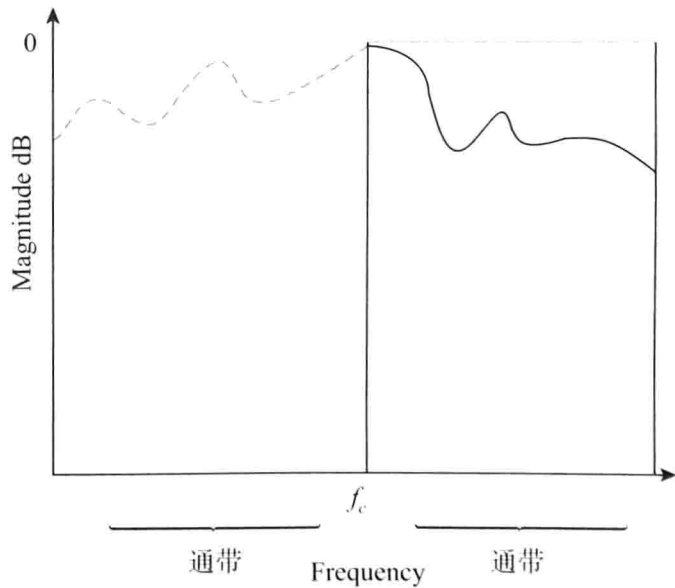


图 2.2 理想高通滤波器

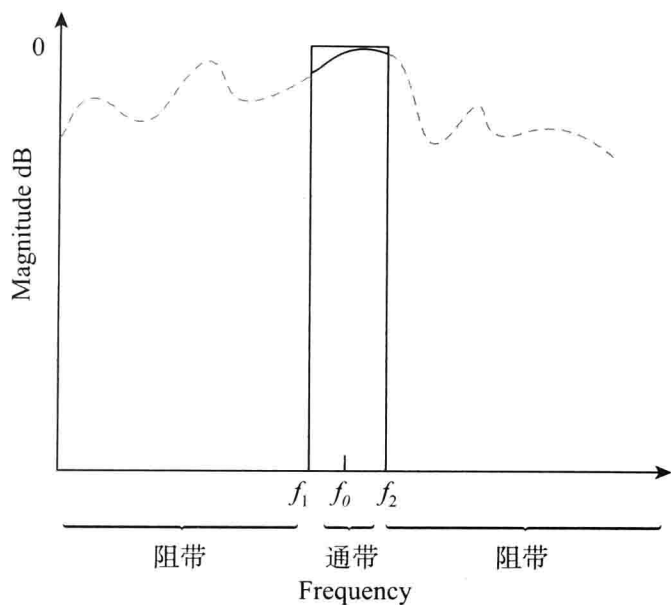


图 2.3 理想带通滤波器

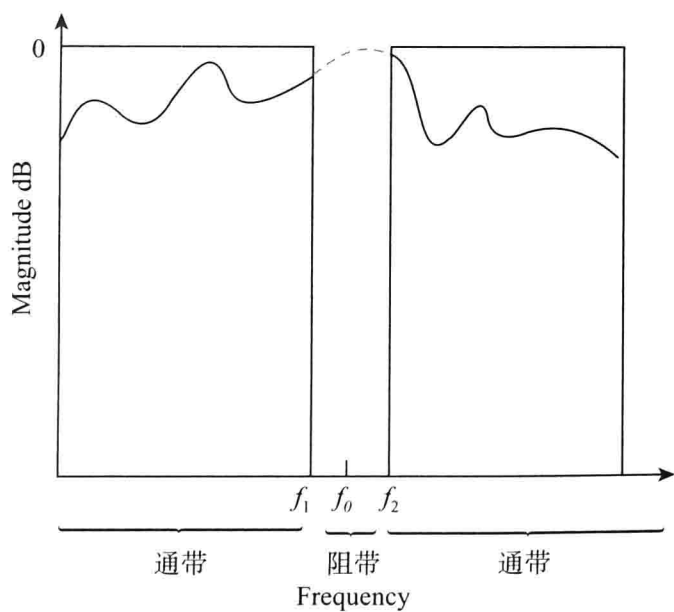


图 2.4 理想带阻滤波器

2.1 滤波器的设计技术参数

本书只会适当地涉及各种滤波器系数算法的细节,对这一部分内容我们将一带而过,而把关注的重点放到滤波器的基本概念和影响滤波表现的技术参数方面。本章也将讨论用于设计满足既定参数滤波器的程序,如 MATLAB[®]。表 2.1 总结了低通、高通、带通和带阻滤波器的主要参数。与理想滤波器不同,实际的滤波器不具备上述墙式滤波器的性能,它们的通带与止带的边界比较平滑,必然会给滤波后的信号带来人为失真。因此,滤波器参数的选择必须要满足输出信号的要求,接下来将进行更详细的阐述。

表 2.1 滤波器的主要参数

参 数	描 述
通带 (Passband)	理想单位增益域:幅值为 0 dB,或非 dB 量纲单位增益 1。
阻带 (Stopband)	衰减域。
过渡带 (Transition Band)	通带与止带之间的区域。
波动 (Ripple)	滤波器所容许的波动量。
滤波阶数/长度/大小/拍 (Filter Order/Length/Size/Tap)	延迟器的数量。

2.2 通带、阻带与过渡带

通带和阻带分别指理想情况下能够通过滤波器和被滤波器截止的频域。理想滤波器在通带具有单位增益且没有失真,在阻带无限衰减。不过实际使用的滤波器在通带和阻带都会有轻微的失真,少量的阻带分量会溢出。如随后 2.4 节的图 2.5 所示,与理想滤波器不同,通带与阻带之间的过渡带 (transition band) 是以平滑或其他方法从通带到阻带的转换带。尽管对过渡带的技术参数没有特别要求,但一般情况下要求过渡带的幅值由通带向阻带单调递减(以保证过渡带总是由通带向阻带递减,而由阻带向通带递增)。音频信号采样率采用 44.1 kHz 而不是 40 kHz 的重要原因之一,是由于过渡带的存在——留下 4.1 kHz 的裕量保证了对于人耳可听频率范围,低通滤波器的上限边缘过渡带是“非砖墙式”的。在理想世界中,这一额外的缓冲量是不必要的,因为人耳听觉系统的频率感知极限是 20 kHz,所以 40 kHz 的采样率就足以满足整个音频信号频域的数字化了。但话又说回来,有时候一些家伙的听觉似乎有点像蝙蝠……这就是需要额外的裕量与比理论需要更高一点采样率的另一个原因。

2.3 截止频率

本节一开始提到的截止频率指通带和阻带开始或结束的频点,它定义为通带边缘幅值下降至 0.707 时所对应的频点(设通带幅值=1)。对于某些特殊的滤波器设计,由于过渡带的存在,截止频率进一步分为通带截止频率和阻带截止频率。这些频率边界也称为过渡带的临界频率 (edge frequency)。

2.4 滤波器的阶数、锐度和纹理

使用数字滤波器时,滤波器的锐度(即与砖墙式滤波器的接近程度)和通带的波动量 (ripple) 是一组重要的参数。过渡带越短小,与砖墙式滤波器或者说理想滤波器越为接近。同时,滤波器的波动量越小(详见下一段),通带越接近于直线(即越接近于理想滤波器),通过频率分量的失真越小。一般说来,滤波器的锐度越高,需要的延迟分量越大(滤波器的阶数越高)。对于 FIR 滤波器,相当于 z 平面上有大量的零点而导致更大的延迟。对于 IIR 滤波器,使用越少的延迟器(滤波器阶数更低),由于反馈分量和共振极点(详见第 6 章)的存在,滤波器的特征就越接近 FIR 滤波器。因此设计滤波器时,需要考虑的重要因素之一是如何在滤波器的阶数与精度之间保持平衡。例如设计一个用于自动检测地震并触发报警的滤波器,会建议使用 IIR 滤波器而不是 FIR 滤波器。因为 IIR 滤波器固有的较小滤波长度使输出结果具有较小的延

迟共序。也就是说,IIR 滤波器具有较好的瞬态响应,对给定的输入信号有较快的响应,如此例中的地震能量——相当于警报系统对地震响应得越早,社区人员撤离的时间就越多(尽管有利的撤离时间很短)。不过,为什么不是所有情况都选择 IIR 滤波器呢?答案在于稳定性是主要因素。对于所有类型的 IIR 滤波器都必须仔细考察其稳定性——因为它们与 FIR 滤波器不同,如第 6 章所述,IIR 型滤波器的反馈分量会影响系统的稳定性,IIR 滤波器不会自动使系统变得稳定。另一个在设计滤波器时需要权衡的因素是如何保持波动量(详见后续内容)与过渡带长度之间的平衡——降低其中一个的值会提高另一个参数的值,就如同阴阳的转换一样。

由图 2.5 和图 2.6 的 FIR 低通滤波器可知,存在一组在砖墙式滤波器中没有的参数,其中包括表征通带幅值摆动程度的波动量(ripple),定义为峰至峰间的幅值距离 A_{pass} ,如图 2.5 所示。波动量这一技术参数有时也称为“容忍度”(tolerance amount),指通带幅值围绕单位增益(0 dB)上下波动的容许量,如式 2.2 所示,其中 δ^+ 和 δ^- 分别为通带幅值响应的正向和负向容忍因子。

$$1 - \delta^- \leq |H(e^{j\theta})| \leq 1 + \delta^+$$

(2.2)

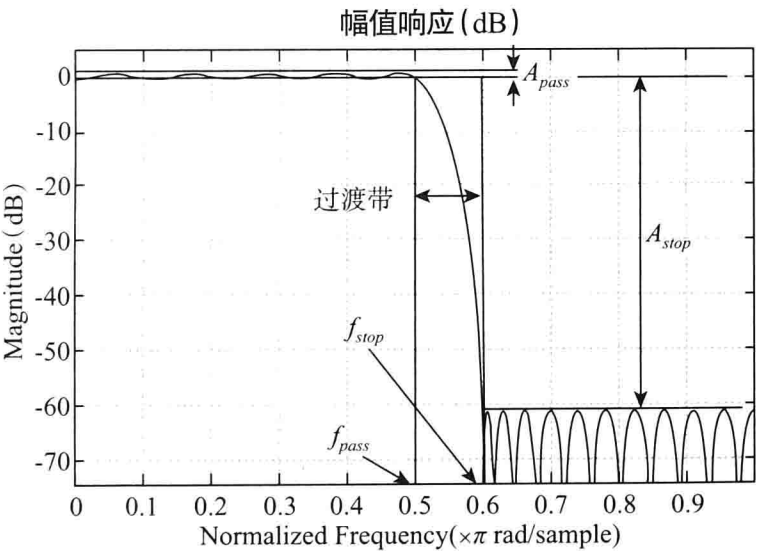


图 2.5 LPF FIR 滤波器的频率响应

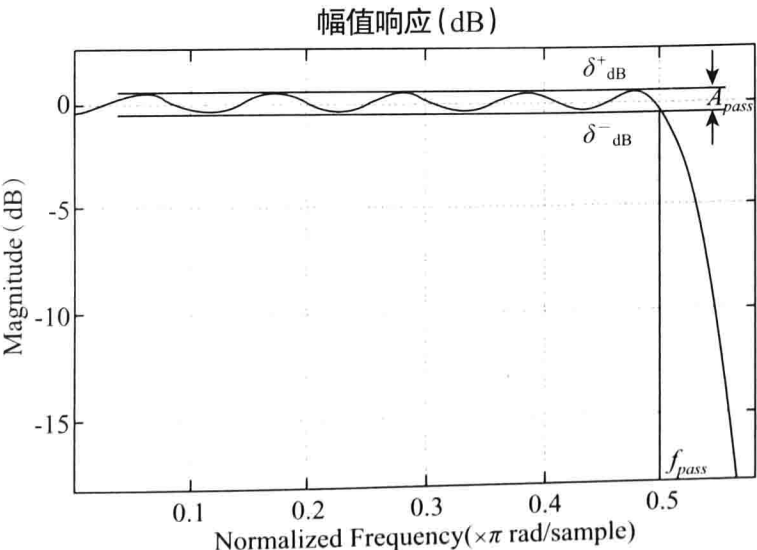


图 2.6 LPF FIR 滤波器的通带波动

我们或许还记得 z 平面上的零点一般有将系统的幅值响应包络拉低至 0 的趋势,而极点则相反——将幅值从 z 平面的极点位置推出增大。这些推挽特征会影响滤波器的波动特性。其他的滤波器技术参数,如观察通带所得到的阻带 A_{stop} 的衰减量 (attenuation amount, 单位通常为 dB), 定义如式 2.3, 输入和对应输出幅值的分别为 A_{in} 和 A_{out} , 阻带和通带的临界频率分别为 f_{stop} 和 f_{pass} 。

$$\text{衰减量} = 20 \cdot \log_{10} \left(\frac{A_{\text{out}}}{A_{\text{in}}} \right)$$

(2.3)

图 2.7 和图 2.8 给出了 FIR 低通滤波的相位响应与带有 42 个零点的 z 平面。后三图,即图 2.9、图 2.10 和图 2.11 为零点个数相同的 FIR 高通滤波器,其幅值响应在较低频段被拉低。

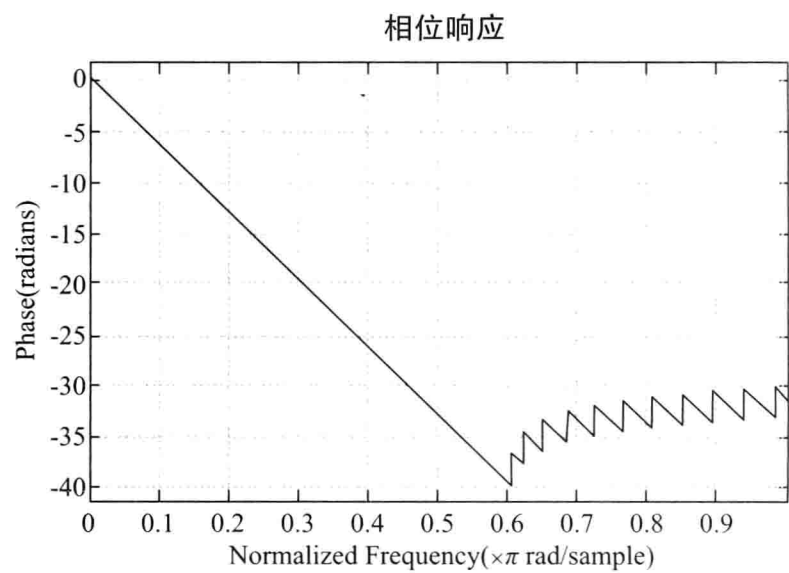


图 2.7 LPF FIR 滤波器的相位响应

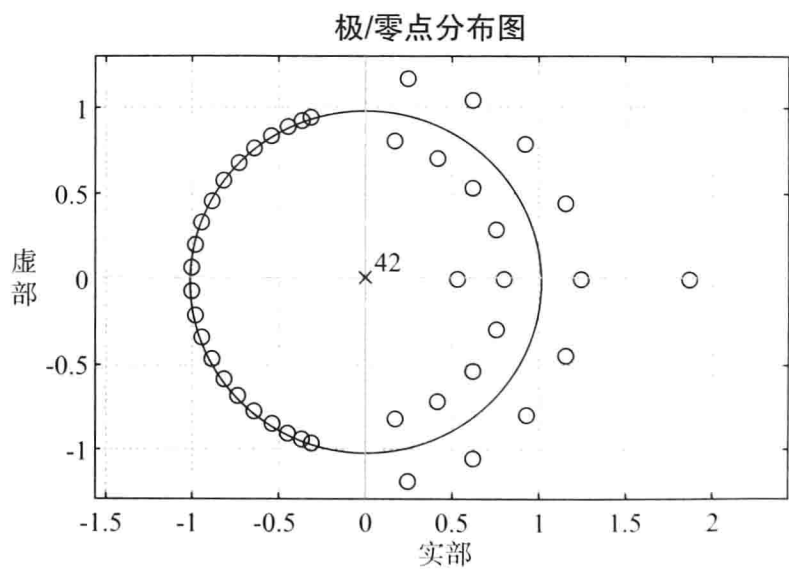


图 2.8 42 个零点的 LPF FIR 滤波器

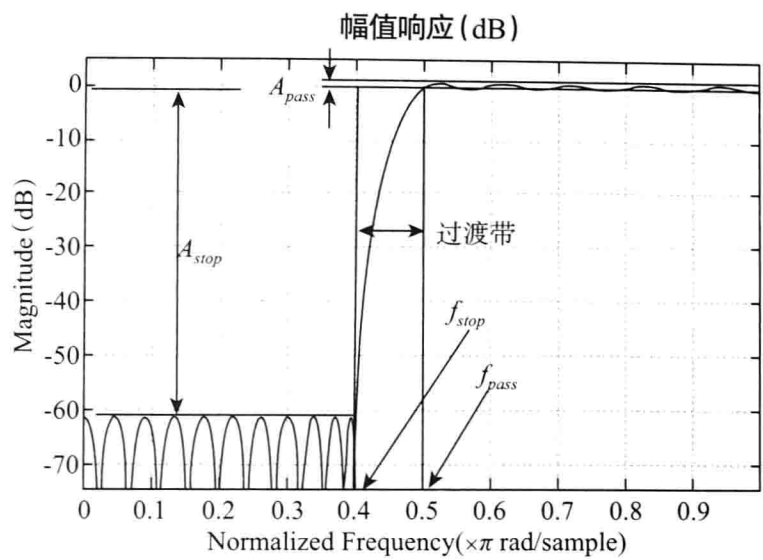


图 2.9 HPF FIR 滤波器的幅值响应

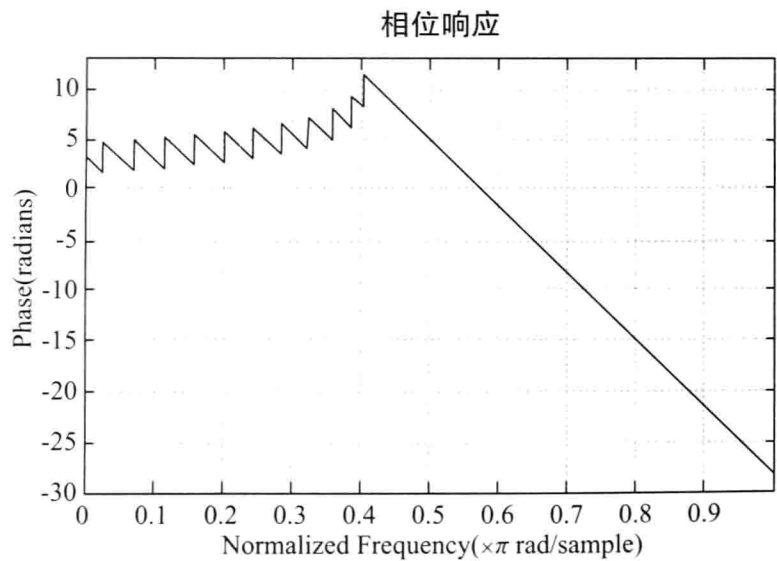


图 2.10 HPF FIR 滤波器的相位响应

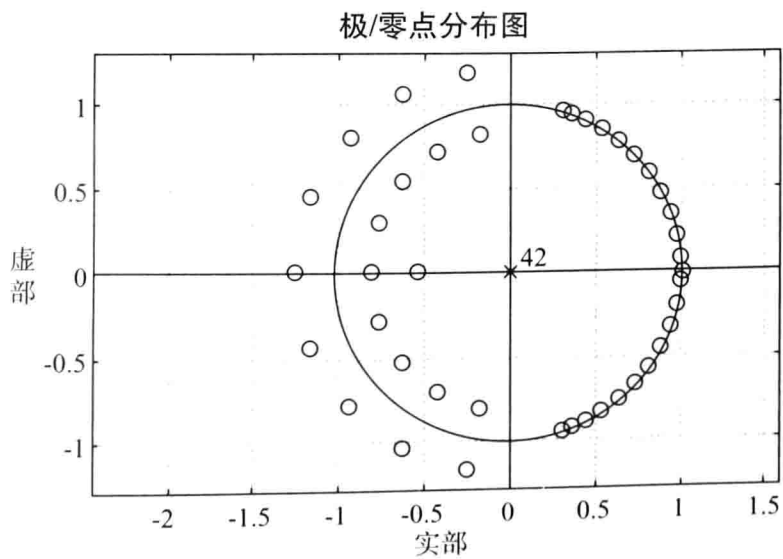


图 2.11 42 个零点的 HPF FIR 滤波器

带通滤波器有一对带通区域,一对通带临界频率,一对阻带临界频率和一个定义为 A_{pass} 的通带范围。带通的锐度则由通带及阻带的临界频率和通带的容忍度定义。其幅值响应、相位响应和带有 44 个零点的 z 平面分别如图 2.12、图 2.13 和图 2.14 所示。

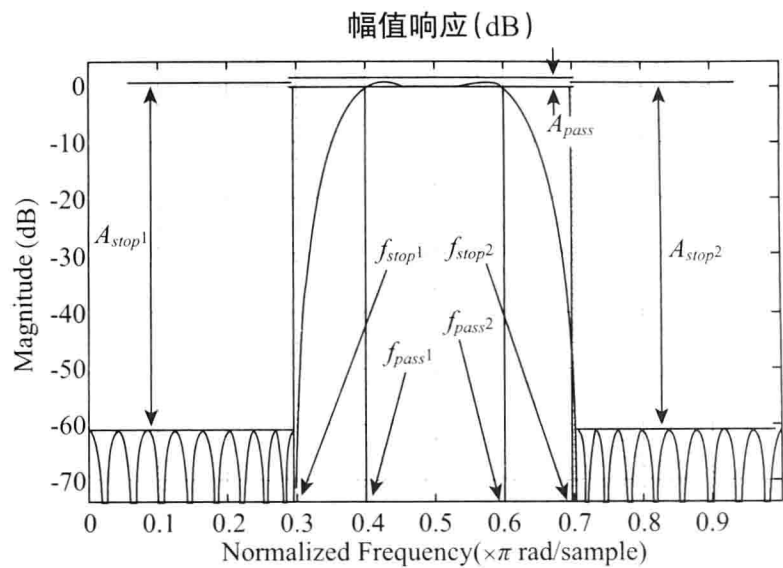


图 2.12 BPF FIR 滤波器的幅值响应

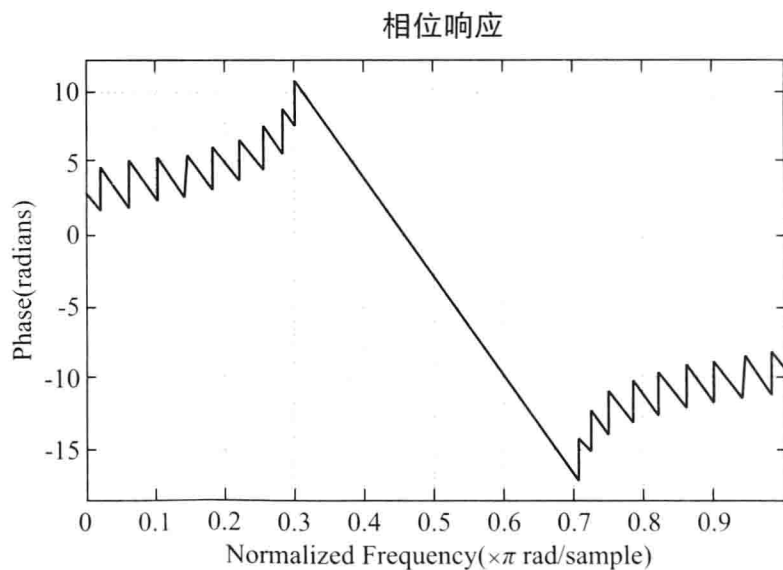


图 2.13 BPF FIR 滤波器的相位响应

另一个量度带通滤波器的通带锐度的参数称为品质因子 (quality factor), 或称为 Q 因子 (Q factor), 又简称为 Q , 定义如下:

$$Q = \frac{f_0}{\Delta f} \tag{2.4}$$

$$\Delta f = f_2 - f_1 \tag{2.5}$$

由式 2.4 可知, 高 Q 值的滤波器带宽窄, 或反过来说带宽 Δf 窄则 Q 值高。宽或窄指以中心频率 f_0 为中心、带宽为 Δf 的频带宽度。 Δf 定义为通带两侧 -3 dB 的频率边界之间的距离, 如图 2.15。

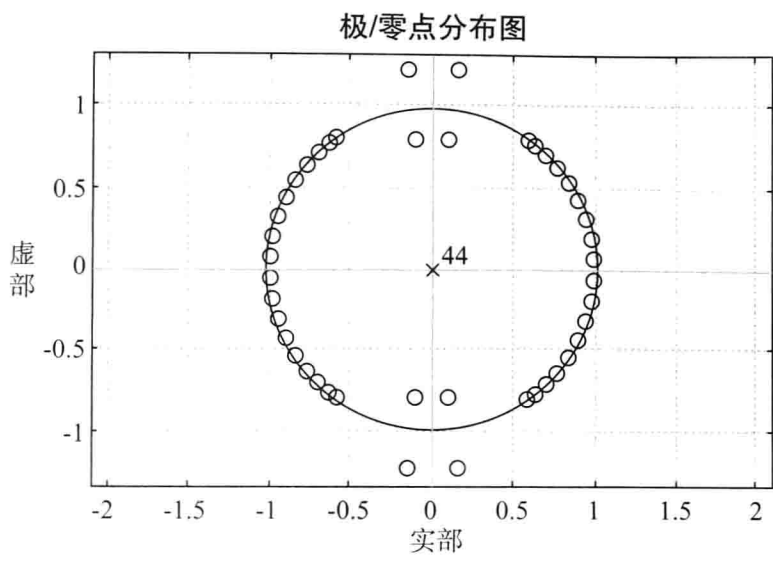


图 2.14 44 个零点的 BPF FIR 滤波器

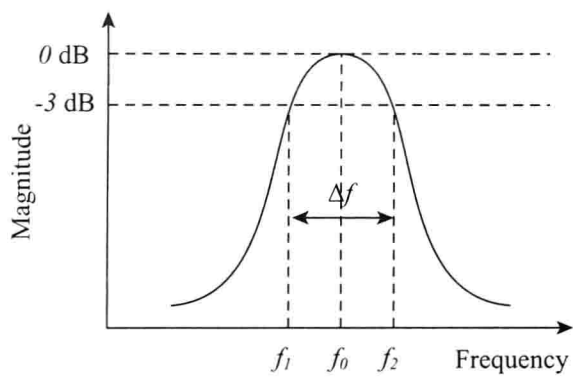


图 2.15 Q 因子与带宽

带阻滤波器的参数如图 2.16。由图可知,该 FIR 带阻滤波器有 40 个零点。其相位响应和 z 平面上 40 个零点的分布见图 2.17 和图 2.18 所示。

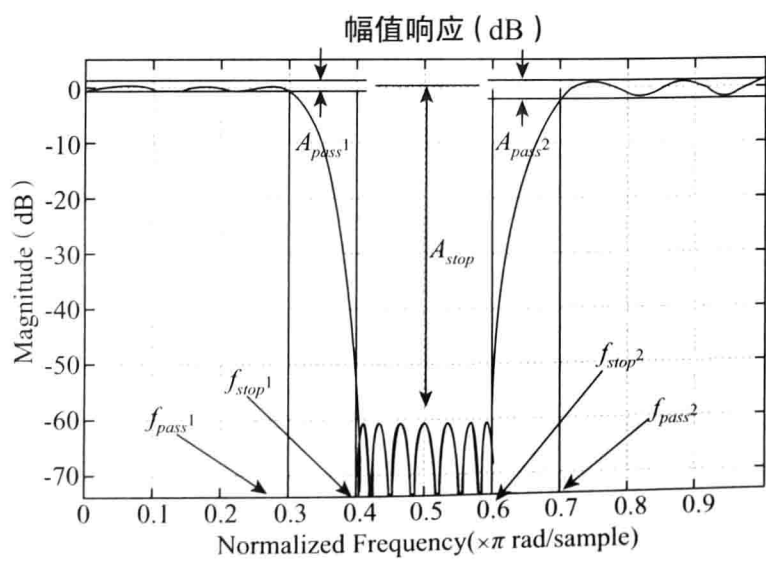


图 2.16 BSF FIR 滤波器的幅值响应

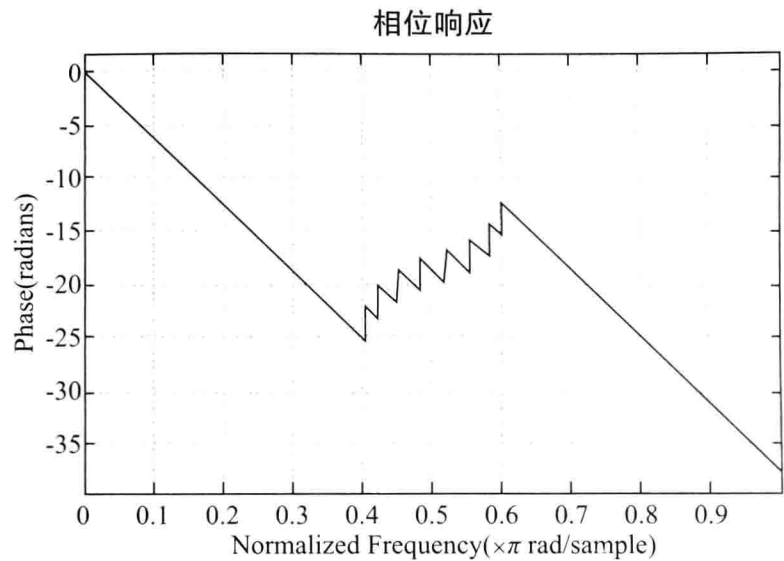


图 2.17 BSF FIR 滤波器的相位响应

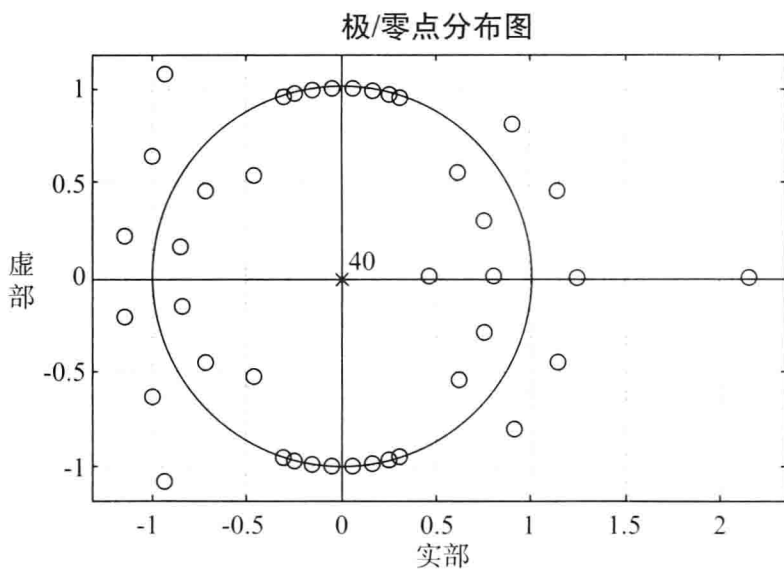


图 2.18 40 个零点的 BSF FIR 滤波器

有多种标准方法可用于通过计算所需要的技术参数来设计滤波器,包括巴特沃斯(Butterworth)、切比雪夫(Chebyshev)和科尔(Cauer)椭圆滤波器算法(这些算法的简要讨论见下节)。巴特沃斯滤波器因具有非常平直的通带和阻带而常用于音乐编辑处理。第一类切比雪夫滤波器则可方便地控制通带和阻带的波动特性,如图 2.19 和图 2.20 绘制的 IIR 低通滤波器。图 2.20 是图 2.19 通带部分的放大图,该滤波器对应具有 10 个零点和 10 个极点的 z 平面,如图 2.21。图 2.22 是一个椭圆滤波器设计方法的运用实例,它在通带和阻带允许有等波动(equiripple,即相同的波动特征),一般是为了满足最低阶数滤波器的需要。

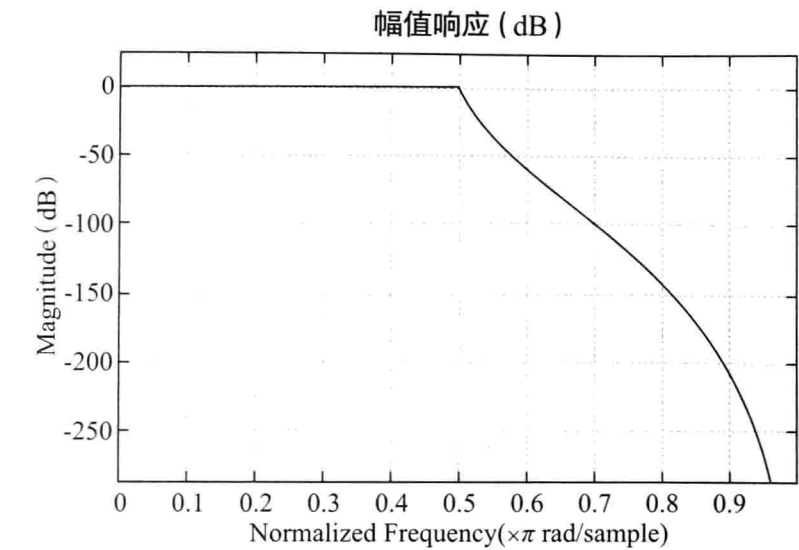


图 2.19 第一类切比雪夫 LPF IIR 滤波器

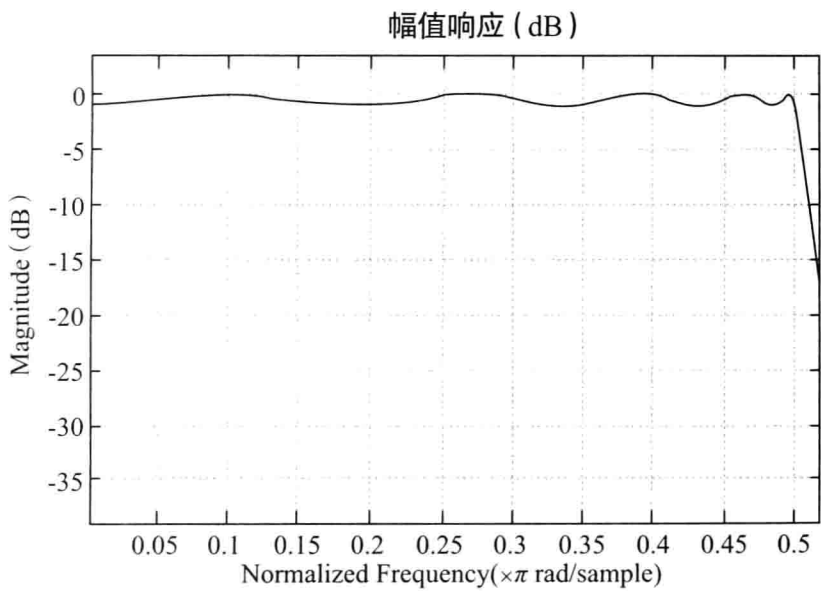


图 2.20 第一类切比雪夫 LPF IIR 滤波器的波动

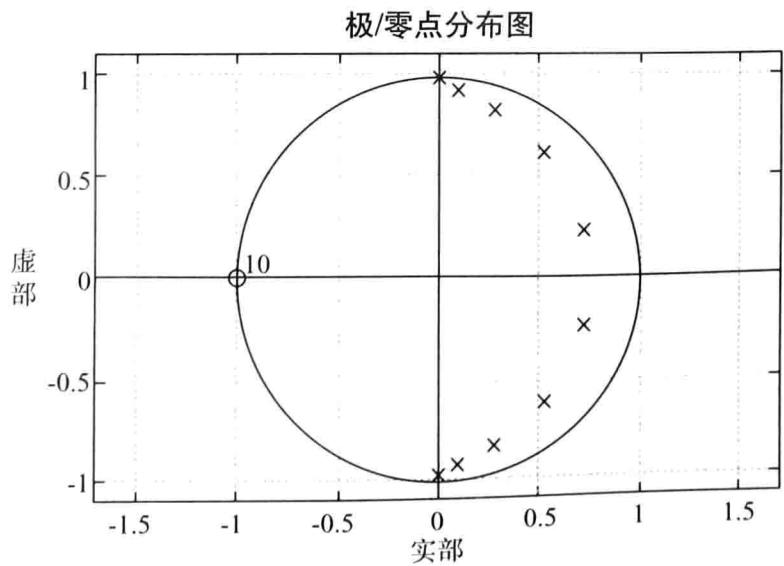


图 2.21 第一类切比雪夫 LPF IIR 滤波器的极零点

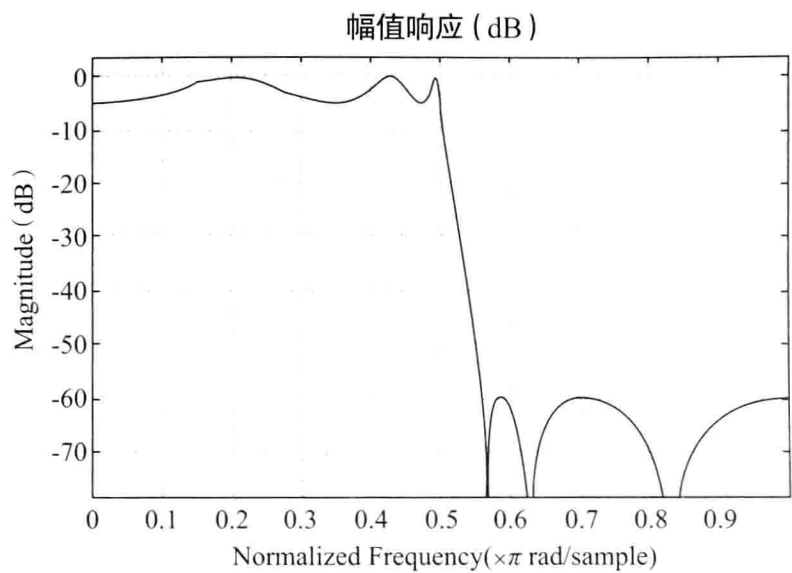


图 2.22 椭圆 LPF 滤波器的准波动

图 2.23 和图 2.24 则是由等波动算法以相同技术参数设计的 FIR 型和 IIR 型滤波器,其中 $f_{\text{pass}} = 0.5$ ($0.5f_s$), $f_{\text{stop}} = 0.51$, $A_{\text{pass}} = 1$ dB, $A_{\text{stop}} = 60$ dB。图 2.25 和图 2.26 给出两个滤波器的零极点结构。由图可知, IIR 滤波器的零极点个数更少, 与 FIR 滤波器有类似的阻带衰减特征。

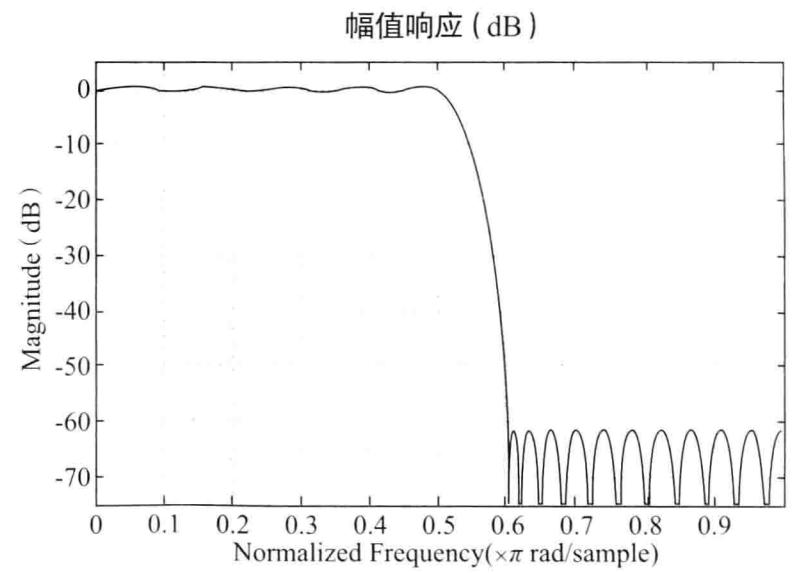


图 2.23 FIR LPF 滤波器

至此,我们生动地认识了具有相同技术参数的各种类型的滤波器,其中 IIR 滤波器具有较短的长度,可短至 6 个极点和 6 个零点,而 FIR 则需要 42 个零点。

2.5 MATLAB® 的滤波器设计工具

如本章一开始所述,由于已经有很多关于计算机编程和数字信号处理的论著对滤波器设计的各种算法和过程做过有效而详细的介绍,本书不会深入讨论各种 FIR 及 IIR 滤波器的设计,只会在简要讨论一些滤波器设计的常用方法之后,介绍用于滤波器设计的 MATLAB® 函数。

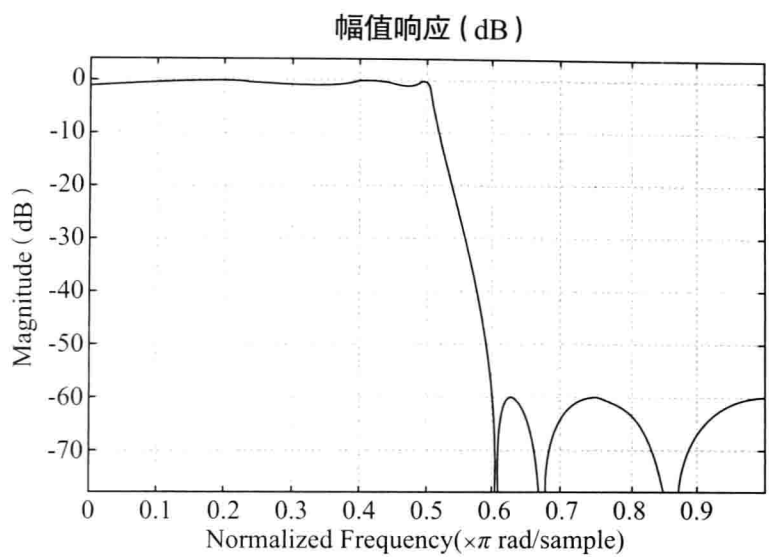


图 2.24 IIR LPF 滤波器

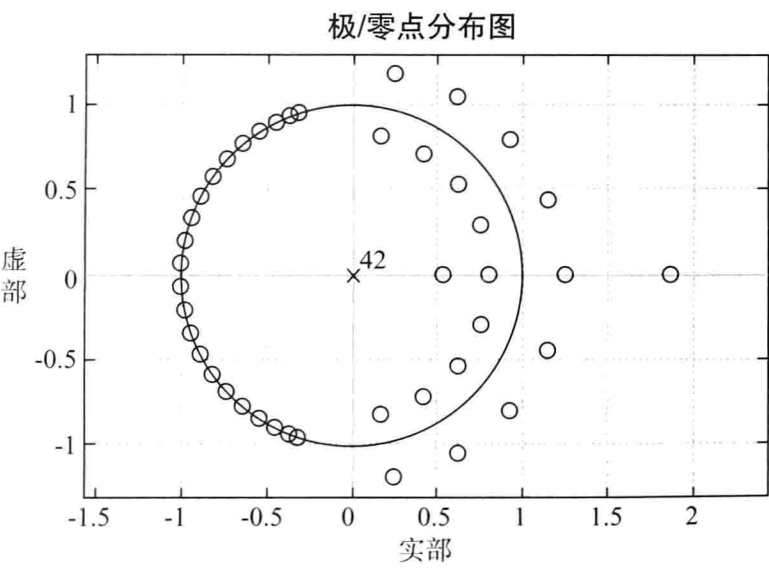


图 2.25 FIR LPF 滤波器的零极点

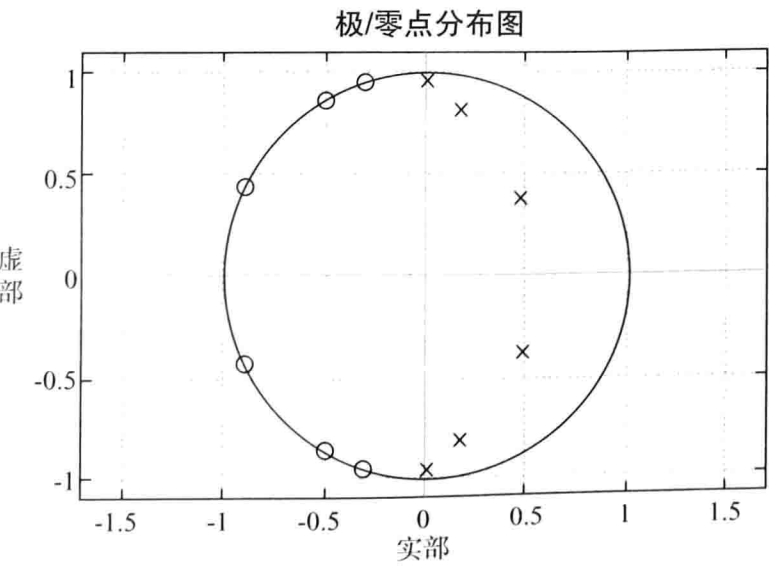


图 2.26 IIR LPF 滤波器的零极点

设计 FIR 滤波器,即无反馈分量的滤波器,两种主要方法分别是脉冲响应截断法 IRT (impulse response truncation method) 和时窗法 (windowing method)。基于截取无限脉冲响应片断的 IRT 法 [波塔 (Porta), 1997] 是最简单的 FIR 滤波器设计方法之一 (理想滤波器具有无限脉冲响应特征)。尽管该方法在结构上非常简单,但它也有负面效应和不利的频域特征,其本身在使用上就有一些不足之处。而时窗法的操作性更强,并可避免由 IRT 法带来的人为失真。顾名思义,时窗法基于选择适当的时窗来设计所需的滤波器 (时窗的讨论及其各种频域特征的介绍详见第 8 章)。总的来说,滤波器的阶数越高,则形状越陡峭——但更高阶的滤波器延迟也越长。不过即使使用时窗法设计 FIR 滤波器,阶数通常也会比较高,从而无法保证滤波器的性能是最优的。因此,一些附加的设计方法如最小二乘法 (least-squares design) 和等波动法 (equiripple design) 常用于优化滤波器的阶数。等波动法可用于控制影响滤波器阶数的有效频带的最大误差。有趣的是,对于 IIR 滤波器,最常见的做法是先设计一个模拟 IIR 滤波器,然后再转换为等价的数字滤波器。最常用的模拟滤波器类型如巴特沃斯滤波器、切比雪夫滤波器和椭圆滤波器。

表 2.2 总结了一些 MATLAB[®] 信号处理工具箱中的函数。对于每一个函数,如果它是一个 IIR 滤波器设计函数,其输出将返回对应传递函数 $H(\cdot)$ 的分子与分母的滤波器系数,而如果它是一个 FIR 滤波器设计函数,则只返回如第 6 章所介绍的分子系数。

$$H(e^{j\theta}) = \frac{\sum_{k=0}^{k=L} b_k e^{-j\theta k}}{1 - \sum_{k=0}^{k=N} a_k e^{-j\theta k}} \quad (2.6)$$

需要注意的是, MATLAB[®] 所输出的滤波器系数均以正权重值 a_k 来表示,则式 2.6 在 MATLAB[®] 中实际上更像式 2.7,因此需要更加注意分母系数的正确极性。比较式 2.7 与式 2.6 可知,其中 a_0 恒等于 1。

$$H(e^{j\theta}) = \frac{b_0 + b_1 e^{-j\theta} + \cdots + b_L e^{-j\theta L}}{a_0 + a_1 e^{-j\theta} + \cdots + a_N e^{-j\theta N}} \quad (2.7)$$

再由差分方程的角度观察该系数可知,在 MATLAB[®] 中,差分方程的反馈分量系数默认为负数而不是正数,如式 2.8 所示:

$$\begin{aligned} a_0 \cdot y[n] &= b_0 \cdot x[n] + b_1 \cdot x[n-1] + \cdots + b_L \cdot x[n-L] \\ &\quad - a_1 \cdot y[n-1] - \cdots - a_N \cdot y[n-N] \end{aligned} \quad (2.8)$$

表 2.2 还列出了每个函数的具体参数和简要描述。一般来说,对于 FIR 滤波器只有反馈前分量,对于 IIR 滤波器则包含反馈前与反馈分量。每个函数返回两个向量 B 和 A 的值,分别对应于式 2.7 的分子和分母部分。参数 W_n 定义为归一化截止频率 ($\pi \times \text{rad/sample}$)——为数字频率。如 W_n 为 0.5 时,对应的数字频率为 $\pi/2$,对于采样率 $f_s = 8 \text{ kHz}$ 的系统其真实频率为 2kHz。由表可知,其中一些函数用于优化给定 IIR 滤波器的阶数——包括函数 $BUTTOR$ ($\cdot$)、 $CHEBYORD1$ ($\cdot$) 和 $CHEBYORD2$ ($\cdot$)。当用户指定通带和阻带频率 (W_p 和 W_s) 以及它们的波动量 (R_p 和 R_s) 时,这些函数返回优化后的阶数 N 。最后, $FIR2$ 函数有两个之前没有提到有趣参数 F 和 A 。因此在 MATLAB[®] 中,命令 $PLOT(F, A)$ 可用来绘制滤波器本身的形状。通常在

这些函数中,频率以 0.0 至 1.0 之间的归一化数字频率表示,其中 1.0 对应的频率为采样频率的一半。

另一个非常有用的 MATLAB® 函数是 FILTER(*B,A,X*) 函数,它用来计算输入信号 *X* 经过系数为 *B* 和 *A* 的滤波器后的输出信号。下节将讨论这些函数的应用实例。

表 2.2 滤波器设计的 MATLAB® 函数

函数名称	类 型	描 述
FIR1(<i>N,W_n</i>)	FIR	生成一个 <i>N</i> 阶滤波器,同时可以使用第三个参数 ‘high’、‘stop’ 或 ‘bandpass’ 来确定滤波器的类型。
FIR12(<i>N,F,A</i>)	FIR	以时窗法生成一个任意形状的滤波器。
BUTTER(<i>N,W_n</i>)	IIR	生成一个 <i>N</i> 阶巴特沃斯滤波器,同时可以使用第三个参数 ‘high’、‘low’ 或 ‘stop’ 来确定滤波器的类型。当 $W_n = [W_1, W_2]$ 为一个二元素向量时,生成的滤波器可以作为一个带通滤波器。
BUTTERORD (<i>W_p, W_s, R_p, R_s</i>)	IIR	返回一个最低的巴特沃斯滤波器阶数 <i>N</i> ,保证幅值衰减在通带处不超过 <i>R_p</i> dB,在阻带处不小于 <i>R_s</i> dB。
CHEBY1(<i>N,R,W_n</i>)	IIR	生成一个 <i>N</i> 阶第一类切比雪夫滤波器,其中 <i>R</i> dB 为峰间最大波动量,同时可以使用第四个参数 ‘high’、‘low’ 或 ‘stop’ 来确定滤波器的类型, W_n 可以为二元素向量。
CHEB1ORD(<i>W_p, W_s, R_p, R_s</i>)	IIR	返回一个最低的第一类切比雪夫滤波器阶数 <i>N</i> ,保证幅值衰减在通带处不超过 <i>R_p</i> dB,在阻带处不小于 <i>R_s</i> dB。
CHEBY2(<i>N,R,W_n</i>)	IIR	生成一个 <i>N</i> 阶第二类切比雪夫滤波器,其中 <i>R</i> dB 为峰间最大波动量,同时可以使用第四个参数 ‘high’、‘low’ 或 ‘stop’ 来确定滤波器的类型, W_n 可以为二元素向量。
CHEB2ORD(<i>W_p, W_s, R_p, R_s</i>)	IIR	返回一个最低的第二类切比雪夫滤波器阶数 <i>N</i> ,保证幅值衰减在通带处不超过 <i>R_p</i> dB,在阻带处不小于 <i>R_s</i> dB。
ELLIP (<i>N, R_p, R_s, W_n</i>)	IIR	生成一个 <i>N</i> 阶椭圆滤波器,其中 <i>R</i> dB 为峰间最大波动量,同时可以使用第四个参数 ‘high’、‘low’ 或 ‘stop’ 来确定滤波器的类型, W_n 可以为二元素向量。
ELLIPORD (<i>W_p, W_s, R_p, R_s</i>)	IIR	返回一个最低的椭圆滤波器阶数 <i>N</i> ,保证幅值衰减在通带处不超过 <i>R_p</i> dB,在阻带处不小于 <i>R_s</i> dB。

3. 滤波器实例

本节将介绍一系列有趣的滤波器专题,包括减法合成、分数延迟、梳状滤波器、弹拨乐器的物理模型和全通滤波器。这些奇妙的方法形成了过去和现在很多计算机音乐应用软件的主干算法。

3.1 减法合成与滤波器

减法合成在 20 世纪 60 年代伴随着广泛使用的 Moog 合成器,以音乐键盘合成器的形式首次提出。简单地说,减法合成就是将一个复杂声音通过滤波得到一个较简单的声音。将声音变得不那么丰满,而是听起来比较奇特,不过当我们开始关注音乐性的结构时,需要那些比“类噪”信号更单薄、更简单的声音。就声音的丰满度而言,可认为白噪声最为复杂也具有最高的丰满度,而理想的单频振荡器产生的声音丰满度最小。减法合成中,一般以具有无调音色的类噪源信号开始进行合成,如鼓一类的打击乐声音,而方波、齿波、角波等也常用来获取有调音色。

在某种程度上,减法合成类似于雕塑过程——就如一开始是一整块无棱角的花岗岩,该无定形物体经雕刻后显现出了明显的形体特征。典型的减法合成设置如图 3.1:

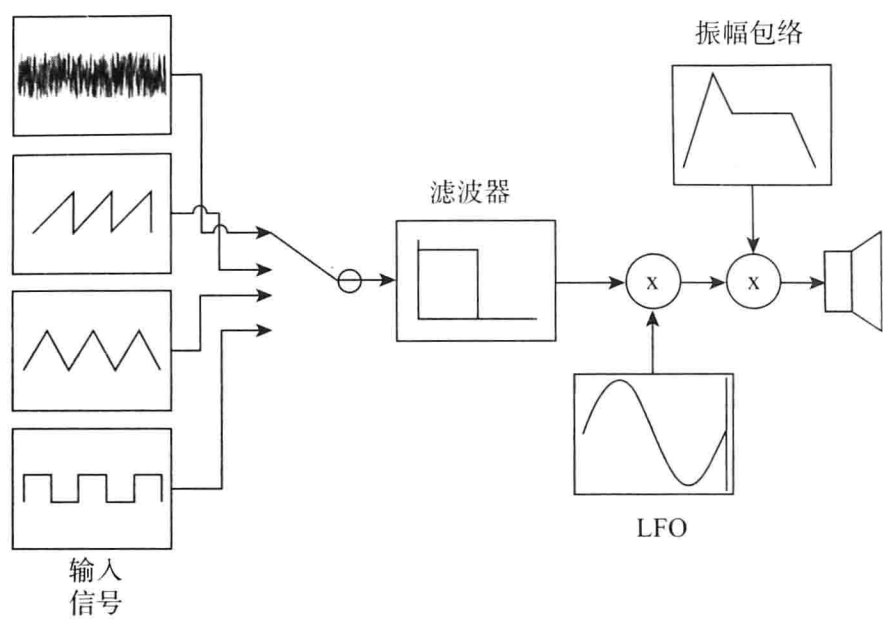


图 3.1 典型的减法合成系统

图中用到了第 2 章讨论的振幅包络生成器与用来进一步增强低频调制的低频振荡发生器 LFO,如对生成信号的振幅调制与音调调制。

以下是一段实施基本减法合成算法的 MATLAB® 代码,它使用了 *RAND*(·) 函数来产生源信号(噪音信号),并用巴特沃斯滤波器设计工具生成一个带通滤波器。代码还采用了指数减的振幅包络,以使生成的信号更接近乐音。该代码还选择使用了由低通和高通滤波器所组成的带通滤波器。

```
fs = 44100;
x = rand (fs,1);           % make random noise
x = x - mean(x);           % take out DC
% make band-pass filter using LPF and BPF
[b1, a1] = butter (10, 0.5);
[b2, a2] = butter (10, 0.4, 'high');
% filter using above coefficients
y1 = filter (b1,a1,x);
y2 = filter (b2,a2,y1);
% envelope the filtered signal
env = 1./exp ((0:(length(y2)-1))/fs).^10;
% play the signal
sound (env'.* y2, fs)
```

代码 3.1 基本的减法合成实例

上例中,*RAND*(·) 函数生成了 44,100 个采样点,或者说时长为 1 秒的信号(采样率 $f_s = 44.1\text{ kHz}$)。第 3 行代码消除了信号的 DC 分量,以保证信号 x 的幅值以 $y=0$ 轴为中心振荡。*BUTTER*(·) 滤波器设计函数生成了两个 10 阶(延迟量为 10)的截止频率,分别为 0.5π 和 0.4π

的低通和高通滤波器,然后叠加合成一个带通滤波器。使用其滤波器系数时,先以低通滤波器对信号进行滤波,一次滤波后的结果再经高通处理。最终滤波后信号的振幅包络特征更明显,也更接近现实的音乐音质。

3.2 双二次滤波器与颤音滤波器

音乐应用中另一个简单而精彩的滤波器范例是“哇哇”颤音滤波器(wah-wah filter)。该滤波器经常被电吉他手使用,有时电贝斯手和键盘手也会使用。这些例子在吉米·亨德里克斯(Jimmy Hendrix)或一些重放克(funky)乐队的演奏中也能听到。最流行的吉他“哇哇”颤音踏板(一种通过踩踏控制乐器声音效果的小金属盒)是 Jim Dunlop Cry Baby 踏板。之所以称为“哇哇”颤音踏板[音乐家也常称之为“踩箱(stomp box)”],是因为如电吉他这样的声音信号经过该滤波器时会制造出类似婴儿啼哭的“哇—哇—哇……”声。这些“短哇哇声”(wah-wah-ness)或“长哇哇声”(wwwaaahhh-wwwaaahhh-ness)通过一个脚踏电位计由人工控制,也有不需要脚踩踏板而自动控制的“自动哇哇”(auto-wah-wah)颤音踏板。尽管这种音色和音乐效果很有趣,甚至也许听起来很复杂,但这种流行效果器背后的原理实际上直观而易于理解。这种“哇哇”声的理论基础仅仅是不断变化带通滤波器的中心频率——对的,就是这样。真的非常简单,不是吗?一种使带通滤波器动态变化(改变中心频率)的方法是采用流行的双二次曲线滤波器(bi-quad filter)。它是一个仅有两个极点和两个零点的 IIR 滤波器,可用于设计高通、低通、带通和带阻滤波器。本节将用双二次滤波器来设计一个带通滤波器以制造“哇哇”效果声。

顾名思义——“哇哇”颤音滤波器得名于其可通过传递函数构造双二次曲线滤波器或与其等价的快速双二次滤波器。双二次滤波器的典型差分方程如式 3.1,对应的传递函数如式 3.2:

$$y[n] = x[n] + a_1 \cdot x[n-1] + a_2 \cdot x[n-2] - b_1 \cdot y[n-1] - b_2 \cdot y[n-2] \quad (3.1)$$

$$H(z) = \frac{1 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2}}{1 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}} \quad (3.2)$$

上式的分子和分母分别乘以 z^2 后,式 3.2 可改写成一般形式,如式 3.3:

$$H(z) = \frac{z^2 + a_1 \cdot z + a_2}{z^2 + b_1 \cdot z + b_2} \quad (3.3)$$

由式 3.3 可知,分子和分母都是二次多项式(二次方程指二阶多项式方程),即“双二次多项式”或“双二次曲线”由此得名。接下来通过求解分子和分母的根将上述传递函数写成更简洁的形式。该分子和分母的根的一般形式分别如下式 3.4 和 3.5:

$$z = \frac{-a_1 \pm \sqrt{a_1^2 - 4 \cdot a_2}}{2} \quad (3.4)$$

$$\begin{aligned} z &= \frac{-b_1 \pm \sqrt{b_1^2 - 4 \cdot 1 \cdot b_2}}{2 \cdot 1} = \frac{-b_1 \pm \sqrt{b_1^2 - 4 \cdot b_2}}{2} \\ &= \frac{-b_1}{2} \pm \sqrt{\left(\frac{b_1}{2}\right)^2 - b_2} \end{aligned} \quad (3.5)$$

由分母的二次多项式的根可知,极点反映了双二次滤波器的共振特征。上述方程的一般解的复数形式如式 3.6。对于实数解,则显然无虚部。

$$z = A \pm jB \quad (3.6)$$

如果方程的根包含虚部(平根号内的值为负数),则根表示为第 6 章讨论的共轭对,详细内容也可见本章先前在 2.4 节中的所述。由欧拉公式知,式 3.6 可写成极坐标的形式如下:

$$z = R \cdot e^{\pm j\theta} \quad (3.7)$$

R 是由实部和虚部定义的欧氏距离,如下式,相位如式 3.9:

$$R = \sqrt{A^2 + B^2} \quad (3.8)$$

$$\theta = \tan^{-1} \frac{B}{A} \quad (3.9)$$

由第 6 章可知,为保证系统的稳定性,极点必须在单位圆内,则有:

$$R < 1 \quad (3.10)$$

同理,对求解分子的根可得到 z 平面上的零点。因此,式 3.3 的传递函数可由欧拉公式改写式 3.11,其中下标 zr 表示零点, p 表示极点,式中的倒数第二步已经将虚部消掉。

$$\begin{aligned} H(z) &= \frac{(z - R_{zr} \cdot e^{j\theta_{zr}}) \cdot (z - R_{zr} \cdot e^{-j\theta_{zr}})}{(z - R_p \cdot e^{j\theta_p}) \cdot (z - R_p \cdot e^{-j\theta_p})} \\ &= \frac{z^2 - R_{zr} \cdot e^{-j\theta_{zr}} \cdot z - R_{zr} \cdot e^{j\theta_{zr}} \cdot z + R_{zr}^2 \cdot e^{j\theta_{zr}} \cdot e^{-j\theta_{zr}}}{z^2 - R_p \cdot e^{-j\theta_p} \cdot z - R_p \cdot e^{j\theta_p} \cdot z + R_p^2 \cdot e^{j\theta_p} \cdot e^{-j\theta_p}} \\ &= \frac{z^2 - R_{zr} \cdot e^{-j\theta_{zr}} \cdot z - R_{zr} \cdot e^{j\theta_{zr}} \cdot z + R_{zr}^2 \cdot 1}{z^2 - R_p \cdot e^{-j\theta_p} \cdot z - R_p \cdot e^{j\theta_p} \cdot z + R_p^2 \cdot 1} \\ &= \frac{z^2 - R_{zr} \cdot (\cos\theta_{zr} - j\sin\theta_{zr}) \cdot z - R_{zr} \cdot (\cos\theta_{zr} + j\sin\theta_{zr}) \cdot z + R_{zr}^2}{z^2 - R_p \cdot (\cos\theta_p - j\sin\theta_p) \cdot z - R_p \cdot (\cos\theta_p + j\sin\theta_p) \cdot z + R_p^2} \\ &= \frac{z^2 - 2 \cdot R_{zr} \cdot \cos\theta_{zr} \cdot z + R_{zr}^2}{z^2 - 2 \cdot R_p \cdot \cos\theta_p \cdot z + R_p^2} \end{aligned} \quad (3.11)$$

将分子和分母分别除以 z^{-2} 或乘以 z^2 ,则式 3.11 的结果可改写成式 3.12 的形式。

$$\begin{aligned} H(z) &= \left(\frac{z^2 - 2 \cdot R_{zr} \cdot \cos\theta_{zr} \cdot z + R_{zr}^2}{z^2 - 2 \cdot R_p \cdot \cos\theta_p \cdot z + R_p^2} \right) \cdot \left(\frac{z^{-2}}{z^{-2}} \right) \\ &= \frac{1 - 2 \cdot R_{zr} \cdot \cos\theta_{zr} \cdot z^{-1} + R_{zr}^2 \cdot z^{-2}}{1 - 2 \cdot R_p \cdot \cos\theta_p \cdot z^{-1} + R_p^2 \cdot z^{-2}} \end{aligned} \quad (3.12)$$

观察由上述传递函数得到的差分方程可知,上述二阶反馈前分量与二阶反馈分量可生成以下双延迟 IIR 系统写成如下两个部分:

$$\begin{aligned} y[n] &= x[n] - 2 \cdot R_{zr} \cdot \cos\theta_{zr} \cdot x[n-1] + R_{zr}^2 \cdot x[n-2] \\ &\quad + 2 \cdot R_p \cdot \cos\theta_p \cdot y[n-1] - R_p^2 \cdot y[n-2] \end{aligned} \quad (3.13)$$

这个双二次差分方程是半径比 R_z/R_p 及频率比 θ_z/θ_p 的函数。而反馈前及反馈分量的二阶延迟系数 $(n-2)$ 与 θ 无关,这就意味着可单独调整 R_p 来优化频率响应和系统的共振特征而保持 θ_p 不变。由第6章可知,数字频率 θ_p 和 θ_z 分别是极点和零点的频率,对于采样周期 $T(1/f_s)$,两者有如式 3.14 所示关系。因此,可定义频率 f 为中心频率(共振频率),通常用 f_0 表示,如式 3.15(用符号 θ_p 来代替 θ_0 ,以免混淆零点与极点的频率)。

$$\theta = \omega \cdot T = 2 \cdot \pi \cdot f \cdot T \quad (3.14)$$

$$f_0 = \frac{\theta_p}{2 \cdot \pi \cdot T} \quad (3.15)$$

令 R_p 趋向于 1,以提高特定共振频率的增益(为保证系统稳定,绝对值 $|R_p|$ 必须小于 1)。 R_p 同时也可作为一个衰减系数, $R_p=0$ 时,由于没有反馈分量,双二次滤波器将变成一个 FIR 滤波器。

因为零点(位于单位圆内时)一般具有反共振特征,所以在零点位置频率响应曲线有拉挽效应(详见第6章),因此常见的双二次带通滤波器的设计方法是,将在 z 平面的 DC 分量($z=0$)和奈奎斯特限制频率($z=-1$)处设置两个零点,如图 3.2:

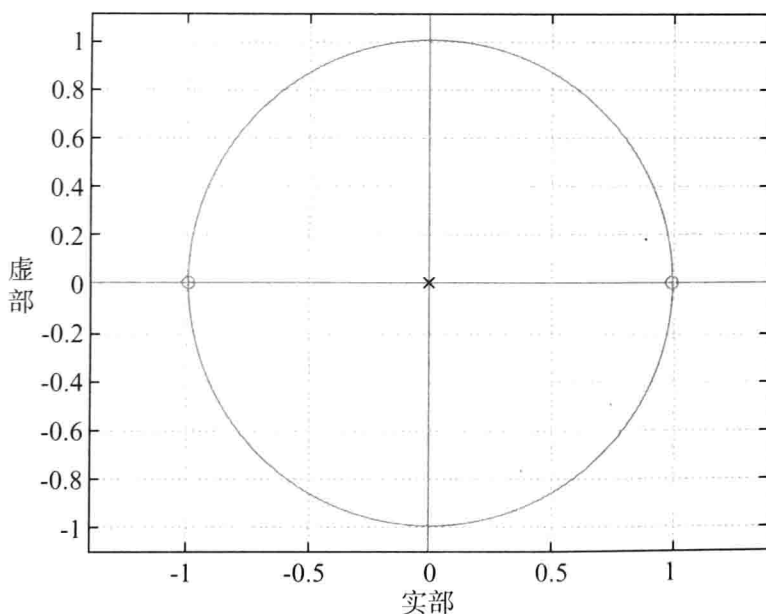


图 3.2 双二次曲线带通滤波器的零点

这样通过设定零点位置可得到一个带通滤波器,需要做的是,分别令 $z=1$ 和 $z=-1$ (式 3.3 的零点位置),并求解传递函数分子的二阶多项式的根。传递函数的分子部分如下:

$$G(z) = z^2 + a_1 \cdot z + a_2 \quad (3.16)$$

零点的根可由式 3.4 求出,再令下式中的 $z=1$ 和 $z=-1$ 。

$$z = \frac{-a_1 \pm \sqrt{a_1^2 - 4 \cdot a_2}}{2} \quad (3.17)$$

得以下两式用于求解未知系数 a_1 和 a_2 。

$$z|_{z=1} = \frac{-a_1 + \sqrt{a_1 - 4 \cdot a_2}}{2} = 1 \quad (3.18)$$

$$z|_{z=-1} = \frac{-a_1 - \sqrt{a_1 - 4 \cdot a_2}}{2} = -1 \quad (3.19)$$

改写式 3.18 与 3.19 求解 a_1 和 a_2 , 可得出反馈前系数。

$$-a_1 + \sqrt{a_1 - 4 \cdot a_2} = 2 \quad (3.20)$$

$$-a_1 - \sqrt{a_1 - 4 \cdot a_2} = -2 \quad (3.21)$$

式 3.20 与式 3.21 相加得式 3.22, 求得 a_1 如下:

$$-2 \cdot a_1 = 0 \rightarrow a_1 = 0 \quad (3.22)$$

将 $a_1 = 0$ 代入式 3.20 或 3.21, 求得 $a_2 = -1$ 。

$$\begin{aligned} -a_1 + \sqrt{a_1 - 4 \cdot a_2} &= 2 \\ \sqrt{a_1 - 4 \cdot a_2} &= 2 + a_1 \\ a_1 - 4 \cdot a_2 &= (2 + a_1)^2 \end{aligned} \quad (3.23)$$

$$a_2 = \frac{a_1 - (2 + a_1)^2}{4}$$

$$a_2|_{a_1=0} = \frac{0 - (2 + 0)^2}{4} = -1 \quad (3.24)$$

则对于传递函数, 其反馈前分量为式 3.25, 分子和分母分别除以 z^2 后, 完整的传递函数如式 3.26。通过观察, 对应系统的差分方程可快速由式 3.26 得出, 如式 3.27。

$$G(z) = (z - 1)(z + 1) \quad (3.25)$$

$$H(z) = \frac{1 - z^{-2}}{1 - 2 \cdot R_p \cdot \cos \theta_o \cdot z^{-1} + R_p^2 \cdot z^{-2}} \quad (3.26)$$

$$y[n] = x[n] - x[n - 2] + 2 \cdot R_p \cdot \cos \theta_o \cdot y[n - 1] + R_p^2 \cdot y[n - 2] \quad (3.27)$$

通过将零点设在 0 和 $f_s/2$ 频点处(即 DC 分量和奈奎斯特限制频率), 可得到一个动态变化的带通滤波器, 如式 3.14 和 3.15 的共振频率 f_0 可变——如此得到一个备用的“哇哇”双二次滤波器。不过在用熟悉的计算机语言生成双二次滤波器代码前, 还需要考虑一个有趣而必要的参数, Q 因子。在此不会切入并钻研 Q 因子及其与带宽关系的细节, 这些内容可用拉普拉斯变换 (Laplace transform) 更好地阐述。而现在要说的是, 参数 R_p 可用于控制带通滤波器 BPF 的锐度, 则 R_p 值越高, 极点增益越大, 带宽越窄(锐度越高), 反之亦然。接下来的四个曲线(图 3.3 和图 3.4)给出了通过如下 MATLAB[®] 代码 3.2 生成的不同 f_0 和 R_p 值的双二次滤波器图形, 零点位于 DC 分量和奈奎斯特限制频率处。


```

Rp = 0.98;           % coeff. for individual
                    % control over Q
f0 = 3000;           % center frequency in Hertz
fs = 8000;           % sampling frequency in Hertz
wT = 2*pi*f0*1/fs;   % analog frequency in radians

% The difference equation coefficients
b = [1 0 -2];
a = [1 -2*Rp*cos(wT) Rp^2];

% frequency and phase response
freqz (b, a)

```

代码 3.2 双二次带通滤波器

另一个与“哇哇”滤波器相似的是相位器效果器 (phaser effect), 它不像“哇哇”滤波器那样通过一个变化的带通滤波器来实现, 而是通过一个动态变化的带阻 (陷波) 滤波器来实现。陷波滤波器在其陷波频率范围附近有严重的相移特征, 未处理信号与已处理信号叠加时会产生不同程度的相位抵消。同样, 通过设置零极点位置可设计动态陷波滤波器。之前的论述已经提供了完成该滤波器设计的工具和方法, 这里不再详述——听起来陷波滤波器的设计是一个不错的项目。

3.3 梳状滤波器

梳状滤波器是一种有趣而富于音乐效果的滤波器。它因其图像形状而得名, 如图 3.5——像一把梳子。和其他滤波器一样, 可以通过 FIR 或 IIR 系统来设计梳状滤波器, 其反馈前分量的差分方程与频率响应如式 3.28 和图 3.5, 其中 L 为延迟量。

$$y[n] = x[n] + b_1 \cdot x[n - L] \quad (3.28)$$

对于 IIR 梳状滤波器, 差分方程如式 3.29 所示, 图像如图 3.6。其峰值锐度由 IIR 系统极点的共振特征决定。

$$y[n] = x[n] + a_1 \cdot y[n - L] \quad (3.29)$$

图 3.7 给出了延迟长度 L 为 11 个采样点时, 不同差分系数对应的 FIR 及 IIR 梳状滤波器的图形范围。由图中可知, 谷 (零点) 与峰 (极点) 的锐度是滤波器系数的函数——系数越接近于单位圆, FIR 或 IIR 梳状滤波器的谷或峰越陡峭, 反之亦然。接下来将用学过的工具对梳状滤波器进行深入考察与阐释。

3.3.1 梳状滤波的含义

由 FIR 系统和 IIR 系统的差分方程, 可知如何得到如图 3.6 和图 3.7 所示的频率响应。在式 3.28 和 3.29 中, 每 L 个采样延迟将在输出 $y[n]$ 上增加“ L 阶延迟量” (L th presence) 并增强。与 FIR 系统不同, IIR 系统将在反馈缓冲延迟链的输入样本中产生衰减的反馈分量, 输出中的每 L 个采样延迟都将成为输入的反馈延迟而增强。因此, 由于该反馈循环的存在, 在 IIR 梳状滤波器中, 每 L 个采样延迟都会被加强, 再加强, 再次加强, 又一次加强……依此类推。如果输出延

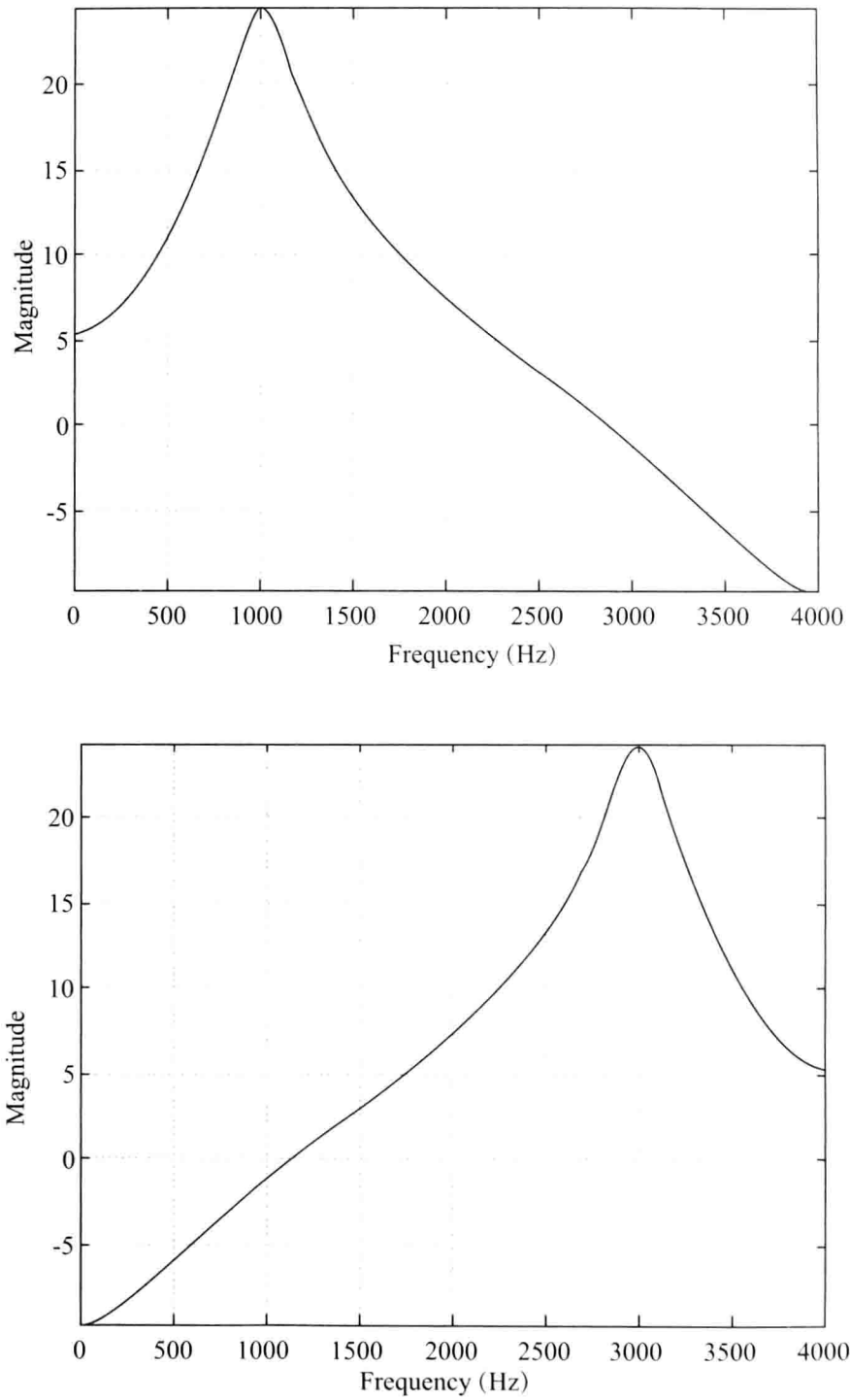


图 3.3 双二次曲线带通滤波器, $R_p = 0.90$, $f_0 = 1 \text{ kHz}$ (上图), $f_0 = 3 \text{ kHz}$ (下图)

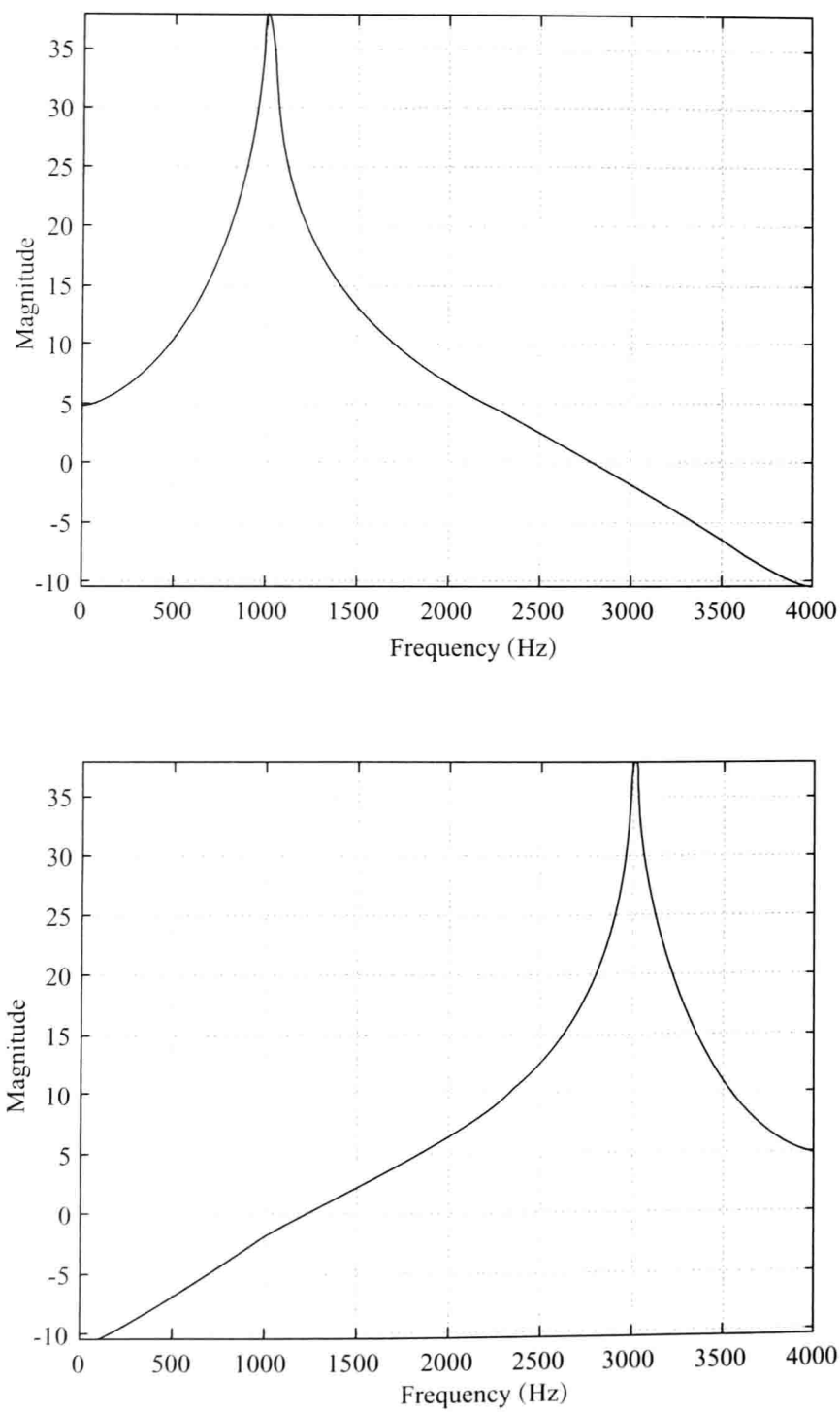


图 3.4 双二次曲线带通滤波器, $R_p = 0.98$, $f_0 = 1 \text{ kHz}$ (上图), $f_0 = 3 \text{ kHz}$ (下图)

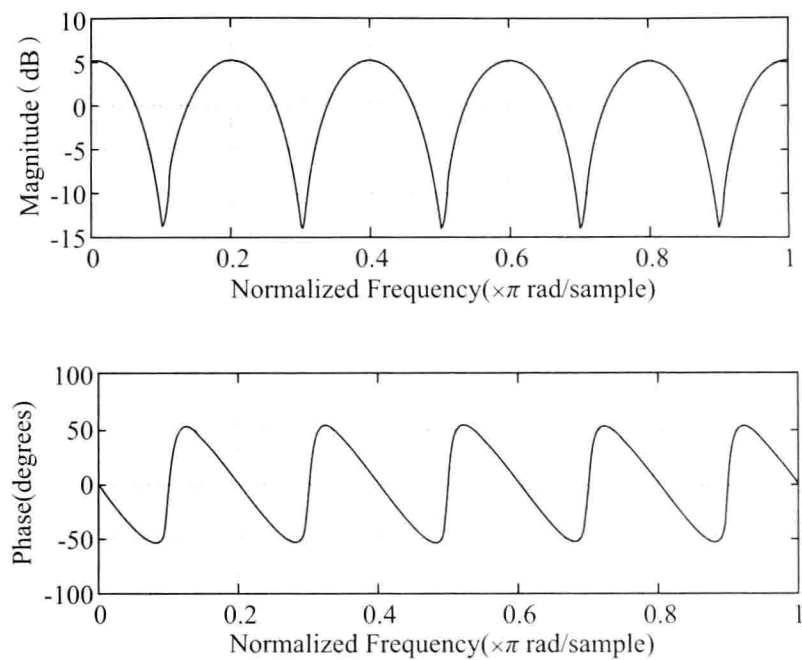


图 3.5 FIR 梳状滤波器, $L=10, b_1=0.8$, 上图为幅值响应, 下图为相位响应, MATLAB[®] 函数代码为 `FREQZ([1 ZEROS(1,8)0.8])`

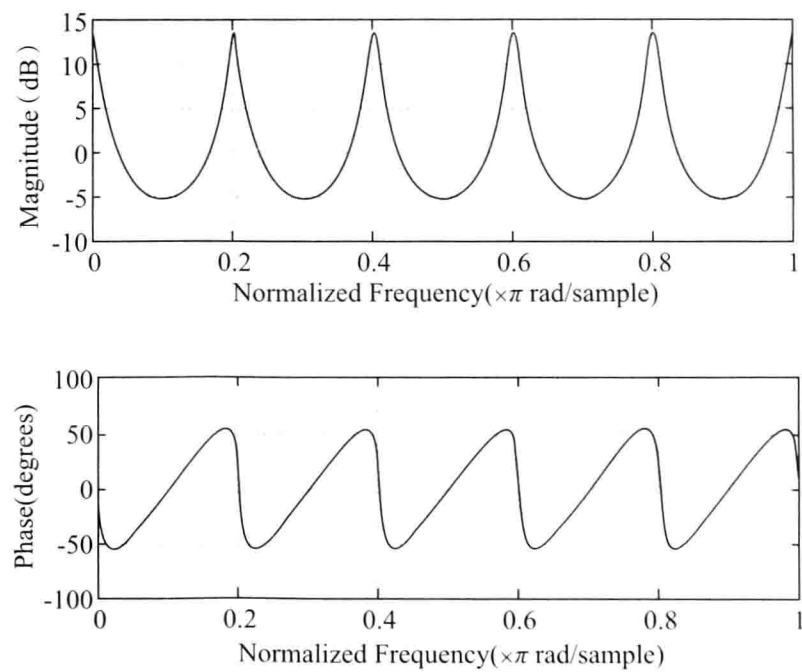


图 3.6 IIR 梳状滤波器, $L=10, b_1=0.8$, 上图为幅值响应, 下图为相位响应

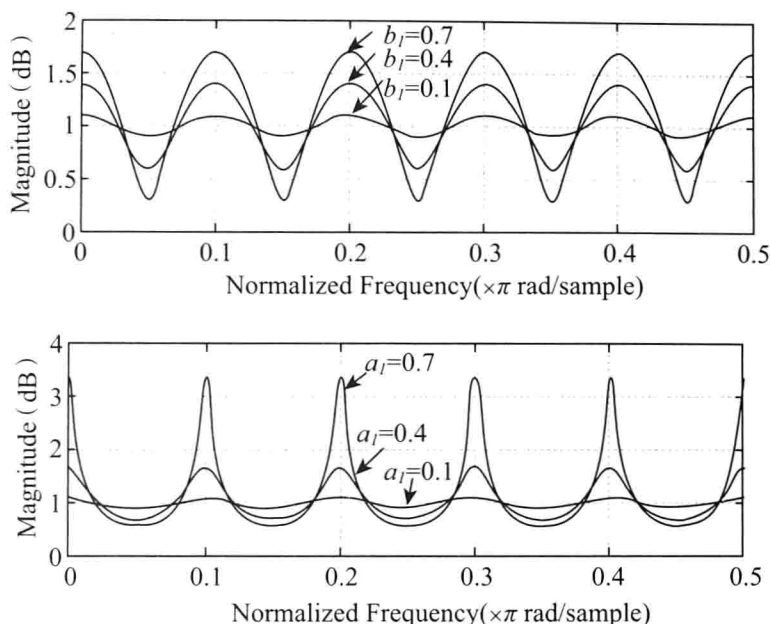


图 3.7 梳状滤波器的频率响应

(系数由 0.1 至 0.9, 步进 0.3, 上图为 FIR 滤波器, 下图为 IIR 滤波器)

迟量在音高范围内——即延迟足够短或足够长, 系统就会产生可感知的有调输出分量。

为了进一步理解梳状滤波器及其因延迟量 L 而产生的采样延迟增强的趋势, 接下来进行脉冲响应分析。对该梳状滤波器输入单位脉冲将出现模式化的响应——脉冲响应序列呈现规则间隔特征, 间隔周期为延迟长度 L 的整数倍, 如式 3.30:

$$\begin{aligned}
 h[0] &= \delta[0] + 0 = 1 \\
 h[1] &= 0 + 0 = 0 \\
 &\vdots \\
 h[L] &= 0 + a_1 \cdot 1 = a_1 \\
 h[L+1] &= 0 + 0 = 0 \\
 &\vdots \\
 h[2 \cdot L] &= 0 + a_1 \cdot h[2 \cdot L] = a_1^2 \\
 &\vdots \\
 h[k \cdot L] &= 0 + a_1 \cdot h[k \cdot L] = a_1^k
 \end{aligned} \tag{3.30}$$

为使系统保持稳定, IIR 系统的系数需满足以下条件:

$$|a_1| < 1 \tag{3.31}$$

因为系数的幂随 L 的整数倍增长, 则该 IIR 梳状滤波器系数的条件可由式 3.30 的结果得证。也就是说, 若 $|a_1|$ 为不小于 1 的值, 该滤波器最终很快会崩溃——即达到系统输出无界增大状态。反之, 若 $|a_1|$ 为小于 1 的值, 该滤波器在某一时刻幅值将达到较小值。这一较小值将会在较短时间内降低为 0, 而使滤波器稳定 (IIR 系统稳定性的进一步讨论详见第 6 章)。因此, 通过对脉冲响应的分析可知, L 阶采样延迟对于 IIR 梳状滤波器的重要性。

接下来, 通过频率响应和 z 变换分析梳状滤波器的零极点特征。首先考虑一个无反馈前分量的 IIR 滤波器, 期望其全极点 z 平面如式 3.32 所示, 令 $|a_1| < 1$ 以保持系统的稳定性。

$$H(z) = \frac{1}{1 - a_1 \cdot z^{-L}} \quad (3.32)$$

为获得极点位置求解分母的根如下:

$$\begin{aligned} 1 - a_1 \cdot z^{-L} &= 0 \\ z^L - a_1 \cdot z^L \cdot z^{-L} &= 0 \end{aligned} \quad (3.33)$$

则极点位置如式 3.34:

$$z^L = a_1 \quad (3.34)$$

为方便计算,令 $a_1 = 1$ (不过这不是 IIR 系统稳定时的情况,但该假设能使数学处理变得更简单),式 3.34 可更简化为式 3.35:

$$z^L = 1 \quad (3.35)$$

由代数学可知,式 3.35 有 L 个根(N 次幂方程具有 N 个根)。通常,为了计算频率响应,将 $z = e^{j\theta}$ 代入式 3.35 可得式 3.36:

$$z^L \big|_{z=e^{j\theta}} = (e^{j\theta})^L = e^{j\theta L} = 1 \quad (3.36)$$

用欧拉公式将式 3.36 展开可知其虚部恒为 0——即式 3.37 的左边没有虚部。另一方面,即上式的结果为实数且必须等于 1。因此,需要找到满足式 3.37 的一组特定 θ 值,则满足该式的解的数字频率 θ 如式 3.38:

$$\cos(\theta \cdot L) + j\sin(\theta \cdot L) = 1 \quad (3.37)$$

$$\theta = k \cdot 2\pi/L, k = 1, 2, \dots, L \quad (3.38)$$

即每当 $\theta = 2k\pi/L$ 时,虚部(式 3.37 的正弦分量)消失,只剩下实部——实部(式 3.37 的余弦分量)如下所示恒等于 1:

$$\begin{aligned} e^{j\theta \cdot L} \big|_{\theta=2 \cdot k \cdot \pi} &= \cos\left\{k \left[\left(\frac{2\pi}{L}\right) \cdot L\right]\right\} + j\sin\left\{k \left[\left(\frac{2\pi}{L}\right) \cdot L\right]\right\} \\ &= \cos(2 \cdot k \cdot \pi) = 1 \end{aligned} \quad (3.39)$$

由式 3.38 可知, $a_1 = 1$ 时, z 平面上的这些数字频率 θ 表征了均匀分布于单位圆上的 L 个极点,相邻极点相差角度为 $2\pi/L$,如图 3.8:

回到式 3.34 可知, $|a_1| < 1$ 时, z 平面上的极点位于单位圆内, IIR 梳状滤波器保持稳定,如图 3.9。同理,对于 FIR 梳状滤波器,将留给读者作为周日下午的数字信号处理作业。需要注意的是, FIR 梳状滤波器的零点恰位于 IIR 系统的极点位置,同样均匀分布于单位圆内,如图 3.10 (其频率响应也可参考图 3.7)。则参数相同的 FIR 梳状滤波器的峰值出现在与 IIR 梳状滤波器相同的频点上,这意味着 FIR 梳状滤波器的零点(与极点的推拉特征不同的是零点有拉挽趋势)必须位于峰值之间(极点位于每个峰值上而零点位于峰值之间)。

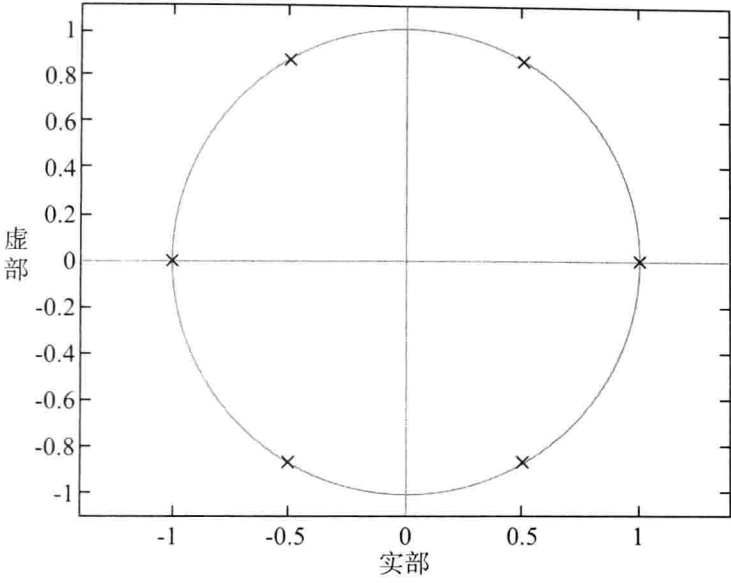


图 3.8 IIR 梳状滤波器的极点分布, $L=6, a_1=1$

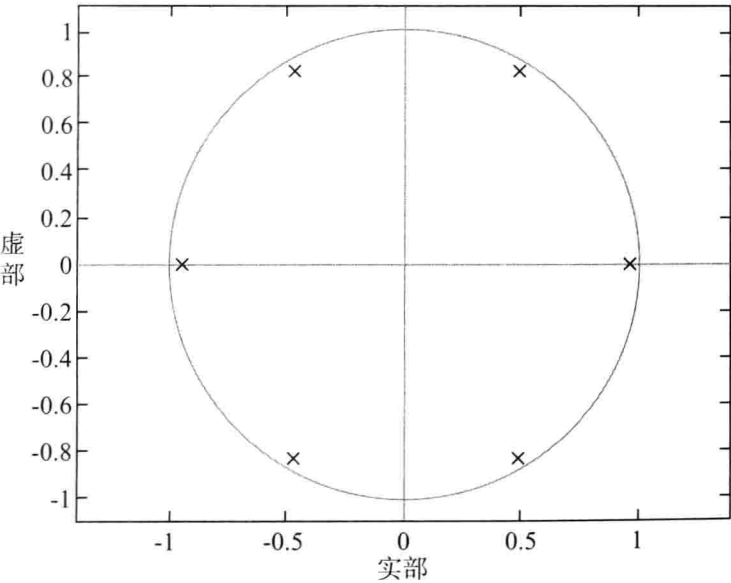


图 3.9 稳定 IIR 梳状滤波器的极点分布, $a_1=0.8, L=6$

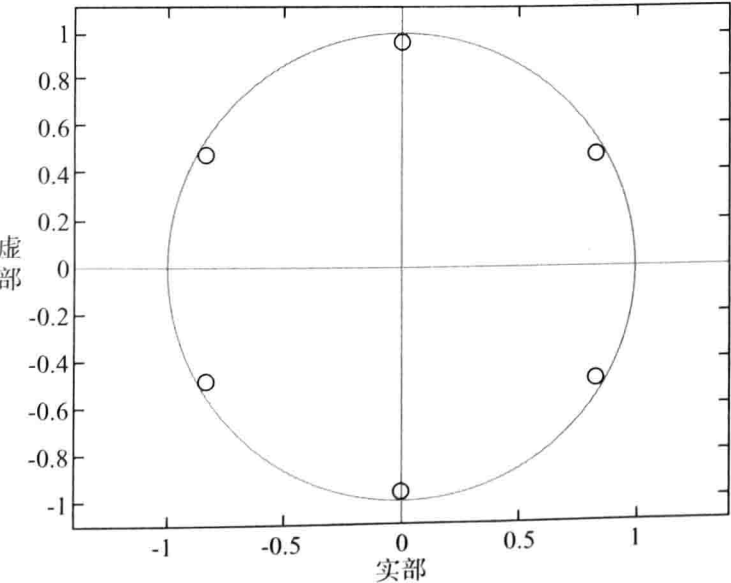


图 3.10 FIR 梳状滤波器的零点分布, $b_1=0.8, L=6$

3.3.2 梳状滤波器举例

梳状滤波很具音乐性的效果,它被音乐团体和音乐技术研究者广泛运用,特别是用无调特征信号由滤波生成有调乐音的时候。一个简单的 IIR 梳状滤波器实例,如 MATLAB® 代码 3.3:

```
% read the audio file
[x, fs] = auread ('loveme22.au');

% set parameters
f      = 440;           % set frequency for comb-filter
L      = round(fs/f);   % set delay length
coef   = 0.99;          % set dampening coefficient

% construct the IIR filter coefficients into an array
% and filter
b = [1 zeros (1,L-1) coef];
y = filter (1, b, x);

% play the resulting sound
sound (y, fs);
```

代码 3.3 基频为 440Hz 的梳状滤波

在以上代码中, MATLAB® 函数 *AUREAD*(·)(或 *WAVREAD*)用于读取音频文件并储存于输入变量 *x* 中。然后通过设置梳状滤波器的参数和频率 *f* 来调整该梳状滤波器并确定相应的整数延迟链 *L*。*ROUND*(·)函数用来对 *L* 进行取整,以保持 *L* 为整数。此例的 IIR 滤波器系数为 0.99,以使滤波器具有较高的锐度,并设置全局 IIR 系数 *b*。同时用函数 *ZEROS*(·)对 IIR 的权重系数进行零填充,以生成适当的延迟链序列。最后用 *SOUND*(·)函数来播放经滤波器处理后的声音。根据所输入的声音文件种类,会听到具有不同音调增强特征的输出信号。可改变频率 *f* 和滤波器系数 *coef*,以及输入信号来感受不同的梳状滤波效果。

接下来将一种特殊的信号作为上述梳状滤波器的输入——这种信号称为白噪声信号。尽管白噪声没有音调特征,但经过该梳状滤波器后将生成有调声。以白噪声为输入的 MATLAB® 代码,如代码 3.4。该代码本质上同代码 3.3,区别在于输入信号 *x* 为白噪声。此处使用 MAT-

```
% set fs and generate white noise signal of length 1 sec
fs = 22050;
x = rand(fs,1);
x = x - mean(x); % get rid of DC offset

% set comb-filter coefficients
f = 220;           % fundamental
L = round(fs/f); % delay length
coef = 0.99;       % IIR coefficient

% build delay vector and filter
b = [1 zeros(1,L-1) coef];
y = filter(1, b, x);

% play sound
sound(y, fs);
```

代码 3.4 噪音的梳状滤波

LAB[®] 函数 `RAND(·)` 生成了时长为 1 秒的白噪声。命令“`x = x - means(x)`”使噪声信号具有 0 均值,以使 x 以 $y=0$ 轴为中心振动(即移除 DC 偏移量)。进一步在输出信号中加入振幅包络,使其特征更接近于乐音的例子,如代码 3.5 所示。

```
% set fs rate and generate white noise of length 1 sec
fs = 22050;
x = rand(fs,1);
x = x - mean(x); % get rid of DC offset

% set comb-filter coefficients
f = 220;          % fundamental
L = round(fs/f); % delay length
coef = 0.99;      % IIR coefficient

% build delay vector and filter
b = [1 zeros(1,L-1) coef];
y = filter(1, b, x);

% create amplitude envelope for output
decay = 8;
expEnv = exp ((0:(length(y)-1))/length(y));
expEnv = (1./expEnv).^ decay;

% envelope output signal
y = y .* expEnv';

sound(y, fs); % play sound
```

代码 3.5 有振幅包络的白噪声的梳状滤波

代码 3.5 生成的声音听起来已经十分真实——这不仅是因为噪声信号由梳状滤波器合成了有调声,而且仅仅加入了简单的振幅包络就能使其更具有音乐性,我敢说,听起来也更加自然。还可以通过改变振幅包络的衰减特征(或尝试不同的包络形状)及基频参数来体验合成后的声音像什么乐器。如果读者一定要做一个猜测的话,这个声音听起来真的像糟糕而明亮的金属弦吉他的声音——这是不是巧合呢? 接着读下去会在下节找到答案。

3.4 弦振动与驻波

可能读者还记得物理课上那个令人难忘的实验,将绳子固定在门的把手上,拿着绳的另一端并绷紧,然后抖动绳子,会看到一个沿绳子传播的波。如果在大脑的长时记忆区域再仔细搜索,还能记得当波动传到门把手时,将以相反的波形反弹回来再传到拿着绳子的手上,如图 3.11。

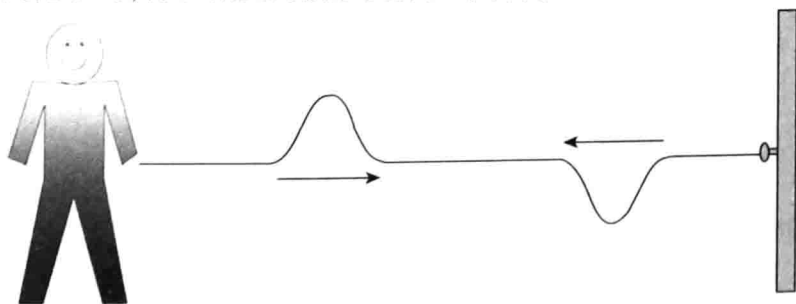


图 3.11 牛顿力学课有关声波反射的实验

如果绳弦振动的能量足够大,一来一回波动传播也许会持续得长一点,在手持一端反弹的倒转波形沿绳传至门把手一端,会使原来的波峰向下,然后在回传过程中与由持绳手传过去的左、右两列行波相遇后叠加,形成位置固定的两个或三个隆起或下陷的波形。这就是弦乐器拨弦的情况,如图 3.12。

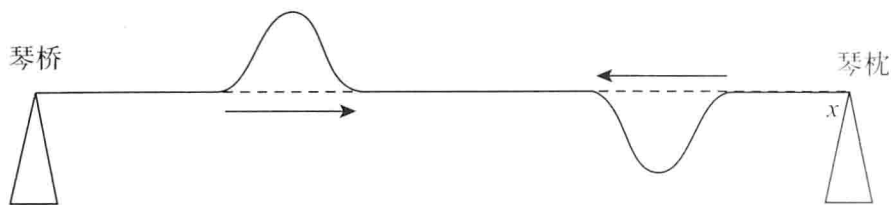


图 3.12 吉他琴桥与琴枕间的弦振动

重要的弦振动数学方程可由两列方向相反的行波叠加表示,如式 3.40。式 3.40 中, x 是弦的位移, c 为行波速度。行波速度可进一步由弦的张力 K 和线质量密度 ρ 表示,如式 3.41 所示

$$y(x, t) = y_l(t + x/c) + y_r(t - x/c) \quad (3.40)$$

$$c = \sqrt{\frac{K}{\rho}} \quad (3.41)$$

回到上述的物理实验,以固定的时间间隔激发绳振动,则向前的行波和回传的行波同时传播而发生干涉,产生“驻波”(standing waves)——即两列在同一弦上传播方向相反的两列行波叠加时,波动实际上不再“行进”,而是保持静止,且振动方向与弦垂直。

在现实世界中,长度为 $L/2$ 的两端固定弦的固有共振频率为 $k \cdot 2\pi / L$ 弧度/秒(radians/sec),其中 k 为整数。单弦振动驻波产生的多种振动模态以及对应波节的位置,如图 3.13。其中,第一类振动模态的频率等价于基频,第二类振动模态在弦中央有一个波节,振动速度是第一类振动模态的两倍,相当于比一阶振动频率高一个八度。波节是驻波的一个有趣特征,假设能够精确地接触任何驻波振动模态的波节,会发现,该节点不会影响任何既有的振动模态(当然,从物理的角度看,由于手指与弦之间的摩擦,这种接触不可能没有能量损耗)。反之,如果接触波腹(振动模态中振幅最大的位置),该特定模态将发生阻尼,并使那些在波腹位置具有最大振幅的谐波消失。

此外,还需要注意的是,弦振波长 λ 与弦的长度有关,如图 3.14。对应基频的最低频率振动(一阶振动模态)需要行波在弦上往返一次才能完成一个完整周期(图 3.15)。上述振动模态及其节点位置对于弦乐器非常重要,谐振能量过大会严重影响乐音和音频信号的声音特性与音色。

上述物理实验中,绳与手指或手之间存在不可忽视的摩擦力,这将使振动快速地衰减。然而,弦固定两端(弦乐器中称为琴枕和音桥)的张力也会使左右行进的行波产生衰减和阻尼,以稳定弦的振动。对于理想的弦振系统,根本不会有能量损失,左右行波将在边界处持续地反射。显然,现实中这种情况是不可能存在的,能量主要以声能、热能和摩擦能量损失的形式在乐器的琴桥和琴枕上损耗。上述现象可用适当的数字差分方程表示——读者也许还注意到弦的共振特征与梳状滤波器相似($k \cdot 2\pi / L$)。因此,可用已知物理模型进行建模,由现实物理世界过渡到虚拟世界。

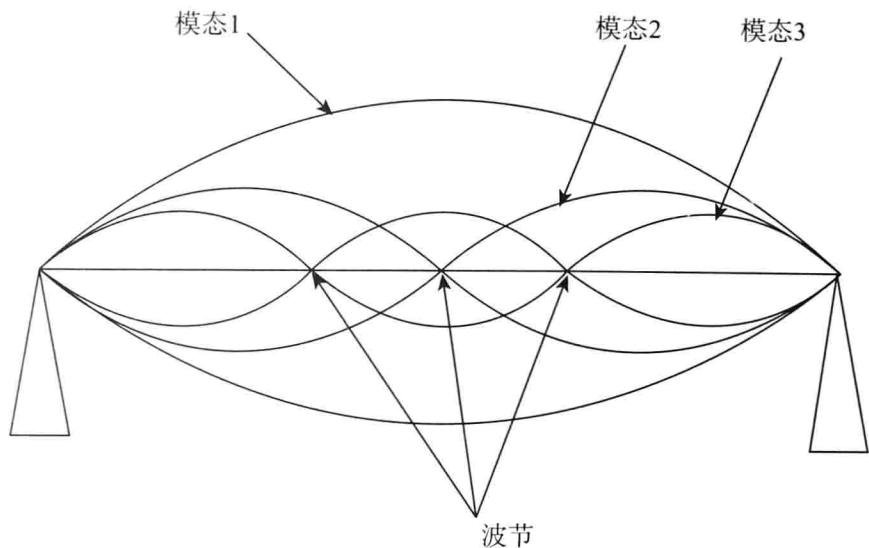


图 3.13 振动模态与对应的波节点

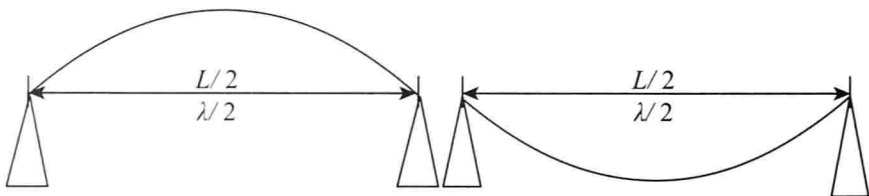


图 3.14 二次谐振的前后两半波形

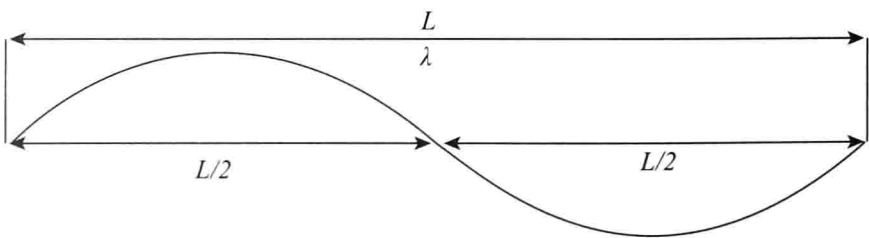


图 3.15 二次谐振的全波形

3.5 物理建模合成与弹拨乐器模型

上节谈到了弦振动的基本特征和弦乐器上左右行波叠加后形成驻波的原理,本节将讨论如何通过“物理建模”(physics modeling)技术来模仿现实物理世界的特征。物理建模合成是指以数学仿真与算法对声学乐器进行建模为主要目标的研究领域。物理建模不仅能够对声学乐器进行建模,其他应用领域还包括模仿模拟合成器和电路,如压缩器、放大器和基于通用真空管的技术也可通过数字模型进行仿真。物理建模合成模型通常也被称为“波导”(waveguide)模型,并在众多研究者特别是来自斯坦福大学的研究人员的努力下,成功地模拟了众多声学乐器。典型的波导模型可用数字滤波器和延迟链构建,以使其在数字信号处理技术上得到理想的应用。

为了获得弹拨乐器模型差分方程和滤波器,可建立弦振动的数字弦振模型。该过程从图 3.12和式 3.40 所示的左右两列行波的数字化开始,如图 3.16:



图 3.16 简化的左右两列行波数字波导模型

由上图所示,将左右两列行波以两个长度为 $L/2$ 的数字延迟链进行替换,并将弹拨弦系统的琴桥与琴枕两个端点用两个方向相反的乘法器进行替换。因为采用了数字延迟链,该系统实际上看起来如图 3.17 所示。图中也为每半根弦(延迟链 $L/2$)增加了能量损失因子,以变量 g 表示,而进一步来自琴枕和琴桥的能量损失分别定义为系数 $-a$ 和 $-b$ 。在声音的产生过程中,能量损耗 g 可认为是由弦与周围空气的内部摩擦而产生的(Smith 1992)。

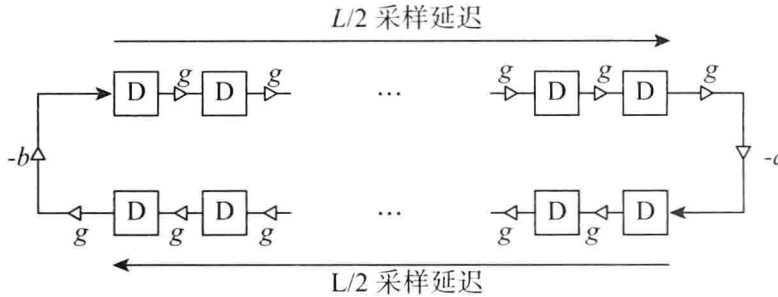


图 3.17 带有单位延迟损耗的数字波导模型

由图 3.17 知,当“数字波”到达每个数字延迟链的端点时,其符号会反相,并以负相乘法器 $-a$ 和 $-b$ 进行阻尼衰减。若系数 a 、 b 和 g 都等于 1,则行波在端点发生全反射,弦振系统本身没有能量损失,行波将永远持续循环。若 $|a| < 1$ 或 $|b| < 1$ 又或 $|g| < 1$,则能量将逐渐损失,导致波动在多次行进后最终停止。由经验可知,吉他弦的振动不会“永远”持续,而会在很短时间内衰减。阻尼因子 a 、 b 和 g 反映了弦振模型的这一属性。

接下来,通过线性时不变(LTI)的交换律进一步将上述系统简化,如图 3.18 所示——即将所有弦的内部能量损耗因子整合为单一变量 g^L :

$$g^{L/2} \cdot g^{L/2} = g^L \tag{3.42}$$

同理,端点阻尼因子也可整合为单一阻尼如下变量:

$$(-a) \cdot (-b) = a \cdot b = c \tag{3.43}$$

则可将能耗因子 g 与 c 统一为单一能耗分量 q ,如式 3.44:

$$q = g^L \cdot c \tag{3.44}$$

最后将两个长度为 $L/2$ 的延迟链合成为一个长度为 L 的延迟链,则该理想 LTI 系统的完整框架如下所示……

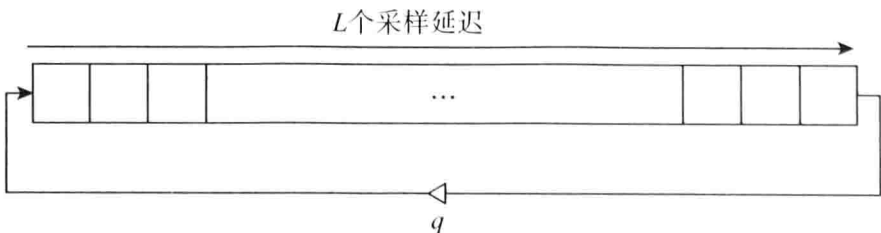


图 3.18 弹拨乐器的统一阻尼模型

接下来,对上述系统进行重组和修改,引入输入 x 、输出 y 以及反馈延迟分量,则系统结构如图 3.19。这是一个读者在本章 3.3 节已经学过并且也已熟悉的滤波器——即反馈式梳状滤波器,其中 $|q| < 1$ 以保持系统的稳定性。

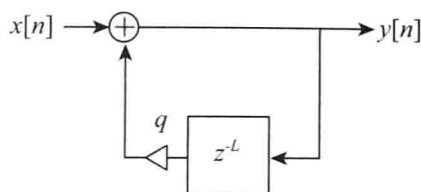


图 3.19 理想弦振反馈滤波器

在第 3.3.2 节,我们已经听到了反馈式梳状滤波器的声音效果,它听起来并不像现实中的弹拨乐器的声音。主要原因之一是,上述理想弦振梳状滤波器模型的所有振动模态具有同样的阻尼。即所有谐波以相同的速率消失,衰减率由参数 q 决定。而这并不是真实乐器的工作情况。事实上,对于真实的弦乐器,高阶振动模态通常的消失速度比低阶要快,基频振动模态往往最后消失。反映这一特征的改进弹拨弦振模型是 Karplus-Strong 弦乐模型 (Karplus, Strong 1983)。凯文·卡普拉斯 (Kevin Karplus) 和阿里克斯·斯朗格 (Alex Strong) 在图 3.19 模型的反馈回路中加入了一个低通滤波器,这一想法非常简单而有效。由于低通滤波器是反馈回路的一部分,反馈信号经过低通滤波器后再返回至输出信号,从而随时间推移抑制了高频分量。这个小小的调整有效地模拟了各弦振模态的不同衰减率——高频衰减较快,而对低频的影响较低。

在 Karplus-Strong 弦乐模型中使用的低通滤波器是最简单的两点平均滤波器,其差分方程如式 3.45。该差分方程将当前采样与其后一点采样进行求和再除以 2——以计算平均值。

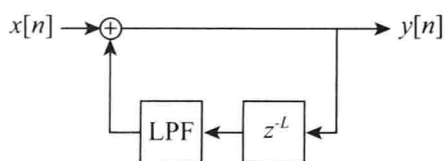


图 3.20 基本 Karplus-Strong 弦乐音合成算法

$$y[n] = \frac{x[n] + x[n-1]}{2} \quad (3.45)$$

最后,使用两点平均滤波器,写出图 3.20 的差分方程,可得如式 3.46 所示结果和基本 Karplus-Strong 弦乐模型的系统框图(图 3.21)。

$$y[n] = x[n] + \frac{y[n-L] + y[n-L-1]}{2} \quad (3.46)$$

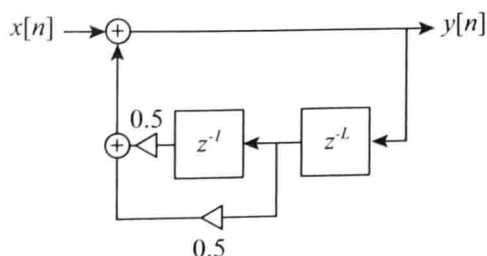


图 3.21 带有低通滤波器的 Karplus-Strong 弦乐模型的系统框图

使用该弦乐模型,以白噪声对弦乐器声进行仿真,并填充和初始化延迟链(像有成千上万个拨片在所有点上同时拨弦),如代码 3.6 所示(将该代码与之前简单的反馈式梳状滤波器做比较,并注意两者的音色差异)。

```
fs = 22050;
len = 2;          % 2 seconds

f = 220;          % fundamental frequency
L = round(fs/f); % equivalent delay length

x = rand(1, L); % make noise: L number of samples

x = [x zeros(1,(len*fs-length(x)))]; % pad 0 to make 2 seconds

a = [1 zeros(1,L-1) -0.5 -0.5]; % feedback coefficients
y = filter(1, a, x);           % filter the pluck

sound(y, fs);
```

代码 3.6 基础 Karplus-Strong 弹弦乐器乐音合成算法

该弦乐模型的幅值与相位响应如图 3.22。由图可知(或可听到)引入了低通滤波器后各谐波具有不同的阻尼特征。由图 3.23 可进一步观察到,z 平面上奈奎斯特极限频率附近的极点有由单位圆向原点靠近的趋势。和所预期的一样,该滤波器的高频分量阻尼更大,而低频分量较小。当然该模型还有许多方面需要调整和改进,由于这些改进算法在许多论文和文章中都有介绍,这里不再一一详述,仅留给好奇的读者作为今后学习的内容。

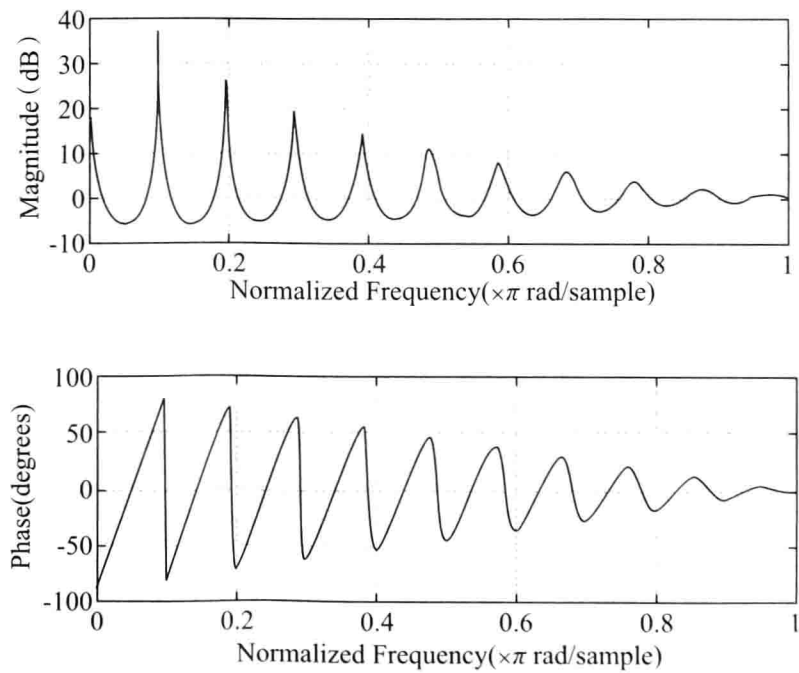


图 3.22 Karplus-Strong 滤波器的频率响应, $L=20$

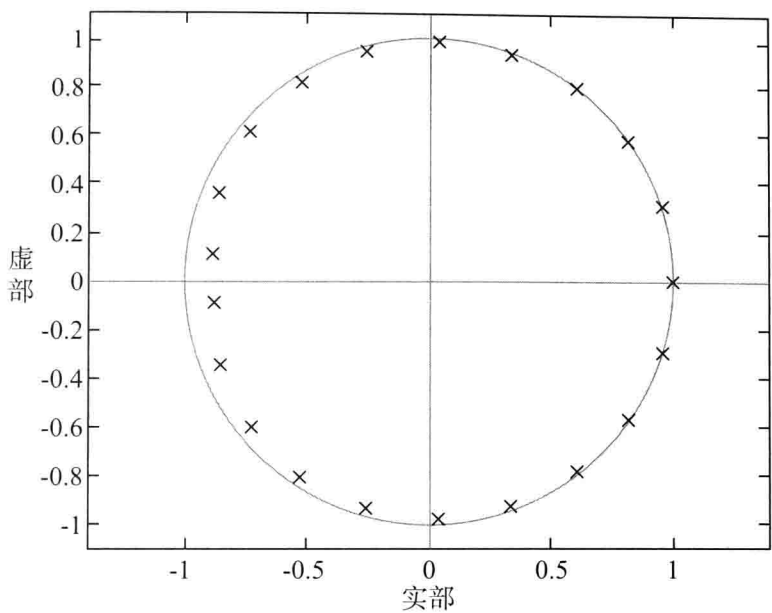


图 3.23 Karplus-Strong 滤波器的平面, $L=20$

在介绍其他滤波器的相关专题之前,在此先简要分析该系统的频率响应,对式 3.46 的差分方程进行 z 变换可得式 3.47:

$$H(z) = \frac{1}{1 - 0.5z^{-L} - 0.5z^{-(L+1)}} \tag{3.47}$$

提取通项延迟分量 z^{-L} 后, $H(z)$ 可改写成式 3.48:

$$H(z) = \frac{1}{1 - 0.5z^{-L} - 0.5z^{-(L+1)}} = \frac{1}{1 - z^{-L} \cdot \left(\frac{1 + z^{-1}}{2}\right)} \tag{3.48}$$

由上式可知,这里已经开始分析图 3.20 弦乐模型——它在结构上是一个带有低通滤波的反馈延迟链。其中, z^{-L} 对应了无损延迟链(频率响应为 1),而 $(1+z^{-1})/2$ 则对应一个表示能耗的平均滤波器,如式 3.49 和图 3.24 所示。

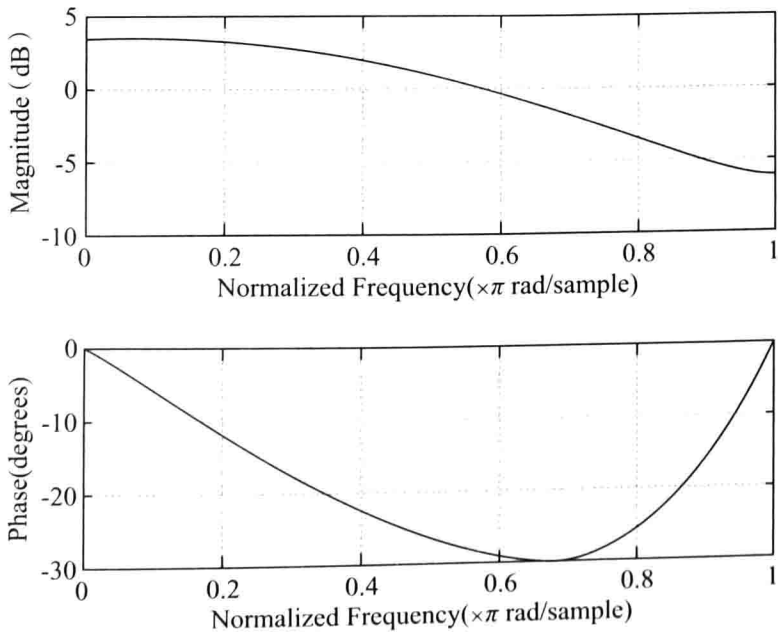


图 3.24 平均滤波器的频率响应

$$H(z) = \frac{1}{1 - H_{delay}(z) \cdot H_{LPF}(z)}$$

(3.49)

在信号处理中,LTI 系统的性质为在现有系统中增加额外的 LTI 数字信号处理模块,以对系统性能进行修改、调整、删除和替换提供了极大的便利。由图 3.22 所示的频率响应可知,该系统是一个被反馈回路中的两点平均低通滤波器染色后的反馈式梳状滤波器。实际上还可通过加入特定的 DSP 模块对原始的 Karplus-Strong 弦乐模型算法进行修改和扩展,以模拟衰变时间的变化、滑奏、更细致的调音(这里提到的基本模型采用的是非分数延迟器,分数延迟器详见 3.6.3 节)和拨弦位置的改变,如图 3.25 所示:

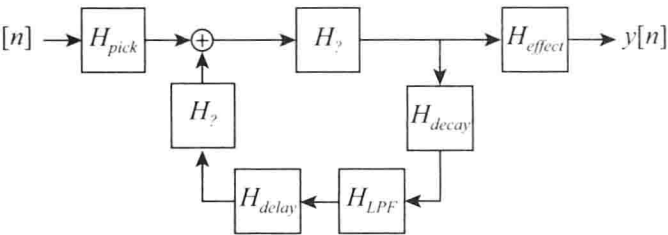


图 3.25 带有附加滤波器的吉他乐音合成模型

该模型也可用来模拟不同类型的弹拨乐器,如木吉他,电吉他,贝斯,以及印度的西塔琴。其中,西塔琴是一个相当有趣的乐器,它拥有一个形状独特的琴桥,并在弦和该琴桥本身之间产生摩擦力,拨弦后会发出一般吉他不存在的“嗡嗡”声,类似于弦打品所产生的嗡嗡声。而对于西塔琴,这种“嗡嗡”声是其音色的重要特点。要模仿西塔琴的“嗡嗡”音色可用波形整形算法对音桥进行建模(Park, Li 2008)。这有助于对之前讨论的常规弦振模态进行失真处理,以获得一种动态变化的嗡嗡声。显然对乐器模型的改进和调整还不止于此,还可对西塔琴的琴身进行建模,添加一个交叉弦振模块,甚至包括如电西塔琴失真的数字效果,并继续做一些功率放大建模处理。

3.5.1 差分方程的通解

至此,可用 MATLAB® 的 *FILTER* 函数构造滤波器和差分方程。而这些函数仅限于在带有相应工具箱的 MATLAB® 环境中使用——如果需要将它移植到一个更通用的计算机编程语言中,如 C/C++语言,则只需对差分方程进行编码即可。例如,编写弹拨乐器模型的程序代码而不使用任何特有的函数和算法,则代码 3.6 可写成代码 3.7。注意到,代码 3.6 和代码 3.7 的真正区别在于,代码 3.7 使用了手动乘法器、加法器和缓冲器,而没有使用滤波器函数。两段代码的执行结果相同。

更仔细地观察代码 3.7 可知,在进行滤波之前,可将延迟链(*yBuffer*函数)初始化为长度为 $N + 1$ 的 0 值数列,使系统成为因果系统,在“电源开关”没有打开前无输出。当然,也可按照需要为弹拨乐器模型的实现设置初始条件。需要记住的重要问题是,在计算过程中要不断更新缓冲区,将旧的缓冲内容及时移除(从缓冲器中删除),然后用最新内容进行替换。在所有缓冲内容被移除后,当前采样成为缓冲器的最后输入,如代码 3.7 [命令 *yBuffer* (1)=*y*(*i*)]。最后,还可通过第四参量 *Zf* 和附加输出分量 *Zi* 来使用 MATLAB® 的 *FILTER* 函数,如代码: [*Y*, *Zf*] = *FILTER*(*B*, *A*, *X*, *Zi*)。利用这些可选参数,可以为滤波器的反馈前和反馈分量设置和检索初始缓冲区(*Zi*)和最终缓冲区(*Zf*)。


```

...
yBuffer = [zeros (1, N+1)];
...
for i = 1: iterations
    % filter signal
    y(i) = x(i) + 0.5*(yBuffer (N) + yBuffer (N+1));

    % update delay line
    % start from back and go towards beginning
    for j = (length (yBuffer)-1):-1:1
        yBuffer (j+1) = yBuffer (j);
    end

    yBuffer(1) = y(i); % store current sample to delay
    buffer
end
...

```

代码 3.7 基本 Karplus-Strong 弦乐合成算法的通用滤波器

3.6 相位在滤波器中的应用

有数不清的声音效果应用于录音、音乐制作、音响设计和音乐作曲以及现场演出中,其中最流行和广泛使用的效果之一是合声效果。它可方便地用 DSP 滤波器实现,用以制造“饱满”(full)和“丰富”(lush)的声音效果,在很多音乐场合非常流行。有时候也用它来叠加人声——当歌唱家缺乏饱满而浑厚的声音,缺乏“现场感”(presence),或常常出现音准问题时,往往会技术性地叠加录制好的人声,通过 DSP 合成技术或手动操控,在现场演唱的同时叠加相同的人声音轨(预先提供原始录制的人声音轨)。后者的人声音轨显然需要与现场原声演唱精密同步,至少可以说这在很多情况下是不容易的(很快就会发现,现实中这是不可能也不可取的)。接下来从第3章4.2.1节简要介绍的合声效果开始讨论相位效果器在音乐中的应用。

3.6.1 合声效果

合声效果的原名很可能来自于其音质效果接近于一个合唱团而不是一个独唱歌手所产生的效果。人声的数量通过人声合奏直接决定了声音的丰满度和丰富程度——因此称其为合声(chorus)。这种效果也被西方传统交响乐文化所证实,一个近100名乐手的军乐团所带来的强大听觉体验不可能由一个乐手或一个小型乐队来完成。交响乐团的弦乐组通常由约20把小提琴、10把中提琴、10把大提琴和6把低音提琴组成。即使所有乐器同时演奏同一个音,也很难准确描述弦乐组同时合奏时所带来的辉煌效果。不过,考虑到演奏家(包括歌唱家)同时演奏(或演唱)时产生的差别,合声效果的人为失真模式也是显而易见的。以两把小提琴同时演奏同一乐谱上相同时长相同音高的音符为例,无论两位乐手的发挥多么完美,总是由于实际操作困难的原因,存在有一些微小的误差使两个音不同步。事实上,同步性会随着齐奏小提琴数量的增加而恶化。这种错误本质上是乐器之间的微小延迟(时域)和由乐器间传声器的校准差异所带来的音准失调(频域)。同样,不管如何努力地校准声学乐器的音准,两个或多个声学乐器

齐奏时也会产生轻微的音准失调。这种失调量取决于乐器的类型、所演奏的音符及其力度、时长和曲目的时长,以及听者与声源的距离。不过这些错误对音乐而言并不一定都是负面影响,因为只要不是特别极端的情况,有时它们可产生丰满甚至“温暖”的音色。当然,对于电声乐器(特别是数字电声乐器)而言,上述音调校准不会有太大问题,而且有时为了获得更自然的听感,甚至会有意让电子乐器的音准失调(添加误差)——以产生合声效果。记得在 20 世纪 90 年代初,基于 FM 技术的合成器雅马哈 V50——利用位于两个 MIDI 通道的相同的电钢琴音色,以几个音分进行失调处理,制造出了非常平滑温暖的电钢琴声音。该合成器合声效果的数字化系统及其基本结构如图 3.26,其原理将在下节详细讨论。

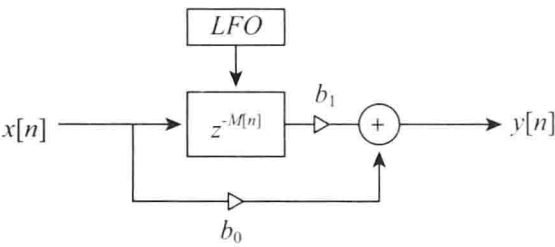


图 3.26 基本的合声滤波器结构

3.6.2 多路延迟滤波器与滤波通道

那么,如何构建数字合声滤波器呢? 如上所述,合声效果的两个主要特征参数是延迟量和失调量。数字合声滤波器使用的延迟器实际上是时变延迟,一组多路延迟结构如图 3.26、图 3.27 和图 3.28 所示。向一系列通道发送延迟的输入人声或声音信号时,每个通道随时间动态变化的延迟量反映了人声的数量。这种可变延迟就会产生合声的失调特征,从而模拟真实乐器间发生的失调效果。值得注意的是,本质上图 3.26 是本章 3.3 节讨论的反馈前梳状滤波器,其差分方程和延迟链 M 如下所示。

$$y[n] = x[n] + b_0 \cdot x[n - M]$$

(3.50)

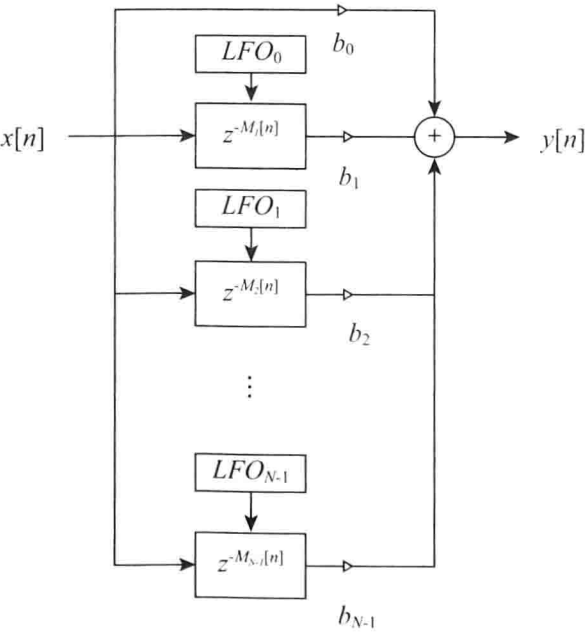


图 3.27 多路滤波器的结构

对于合声滤波器, M 是时间的函数, 通常由一个低频振荡器 LFO 控制。低频振荡器 LFO 的幅值在最小与最大限值间扫动就可控制延迟量, 如式 3.51, 其中 A_{LFO} 为最大振幅, n 为采样点序号, F_{LFO} 为低频振荡器 LFO 的频率, f_s 为采样频率。

$$M[n] = A_{LFO} \cdot \sin(2 \cdot \pi \cdot f_{LFO} \cdot \frac{n}{f_s}) \quad (3.51)$$

一般多路延迟器结构如图 3.28 和式 3.52 所示, 可看成加权输入信号与加权延迟量 $D=1/f_s$ 的输入样本 (FIR 滤波器的输入) 的线性组合, 总最大延迟量为 $(N-1)/f_s$ 。

$$y[n] = b_0 \cdot x[n] + b_1 \cdot x[n-1] + \dots + b_{N-2} \cdot x[n-N-2] + b_{N-1} \cdot x[n-N-1] \quad (3.52)$$

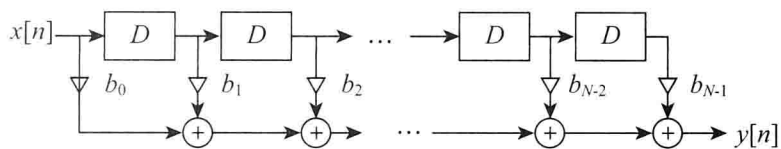


图 3.28 多路延迟链

像图 3.28 这样将每一个延迟分量都叠加至输出中的滤波器结构非常少见, 通常在将结果叠加至输出之前会对模块化的延迟链进行加权处理。任何情况下, 一个动态变化的离散延迟链可放置于图 3.28 的任何延迟位置。例如, 如果需要由第 $(K-100)$ 至第 K 个采样增加一个延迟器, 则可先对第 $(K-100)$ 个采样增加一个延迟器, 然后是第 $(K-99)$ 个采样点, 依次类推直至第 K 个采样。利用这种延迟链, 通常也被称为建立缓冲区, 在合声效果中, 可通过一个基于低频振荡器的索引结构实现。

例如, 若合声效果中的低频振荡器是一个正弦振荡器, 可将其频率设置为较低的值, 如 0.5Hz, 然后将正弦分量的幅值由延迟加入位置至结束位置平滑变化——上例为 $(K-100)$ 至 K 。如下为一段简单的单路合声效果处理器的代码。

代码 3.8 很容易被修改为一个多路合声结构: 以各自的低频振荡器 LFO 和延迟特征添加更多的缓冲区, 再将缓冲区代码加入到主循环中, 则程序如代码 3.9。这两段代码都利用了 MATLAB[®] 中特殊的短代码形式“:”和“end”来更新缓冲区, 以避免像代码 3.7 那样使用“for”嵌套循环来进行搜索。最后, 不论使用什么方法, 不同的系统和计算机语言都有自己的性质、特点和处理特定问题的特殊函数、操作方式以及具备特殊算法的高效运算方式。最终的目标是以所期望的效率获得正确的结果。

3.6.3 分数时延滤波器

由于合声延迟的动态变化特质, 其典型延迟量在 10 毫秒至 50 毫秒间平滑变化, 则需要用分数延迟器实现。如果仅仅使用非分数延迟, 则由采样率 f_s 和整数 K 决定延迟步进, 如式 3.53 所示。这意味着每个延迟增量或延迟衰减量只能是采样周期 $T=1/f_s$ 的整数倍。实现分数延迟的两个常见方法是线性插值法 (linear interpolation) 和全通滤波器法 (all-pass filtering), 本节将介绍第一种方法, 后者将在第 3.6.5 节中讨论。

```

[x, fs] = auread ('loveme22.au'); % read audio file

buffer = zeros(1,1000); % allocate and init. buffer (delay line)
y = zeros(1,length(x)); % allocate and init. output

f = 0.1; % LFO frequency
startIdx = 500; % start delay in samples

sineTable = abs(sin(2*pi*f*[0:length(x)-1]/fs)); % make LFO

% tap index from startIdx to end of buffer
bufferIndex = round(sineTable*(length(buffer)-startIdx))+startIdx;

% run chorus filter
for i = 1:length(x)
    y(i) = x(i) + buffer(bufferIndex(i));

    % update the buffer: shift right (last is oldest)
    buffer(2:end) = buffer(1:end-1); % shift
    buffer(1) = x(i); % update
end

% play the sound
sound(y, fs)

```

代码 3.8 单路合声效果器

```

...
for i=1:length(x)
    y(i) = x(i) + buffer1(bufferIndex1(i)) +...
           buffer2(bufferIndex2(i)) +...
           buffer3(bufferIndex3(i));

    % update the buffer: shift right (last is oldest)
    buffer1(2:end) = buffer1(1:end-1);
    buffer2(2:end) = buffer2(1:end-1);
    buffer3(2:end) = buffer3(1:end-1);

    % update
    buffer1(1) = x(i);
    buffer2(1) = x(i);
    buffer3(1) = x(i);
end
...

```

代码 3.9 多路合声效果

$$\text{delay}_{\text{non-fractional}} = K \cdot \frac{1}{f_s} \quad (3.53)$$

延迟的线性插值可直接实现,如式 3.54,其中 τ 为分数延迟量, $y_{FD}[n-\tau]$ 为延迟后的输出信号, $\text{frac}(\tau)$ 是分数延迟量 τ 的小数部分[尾数(mantissa),即对数的小数部分], $\text{int}(\tau)$ 是分数延迟量 τ 的整数部分。

$$y_{FD}[n - \tau] = a_{\text{int}(\tau)} \cdot y[n - \text{int}(\tau)] + a_{\text{int}(\tau)-1} \cdot y[n - \text{int}(\tau) - 1] \quad (3.54)$$

$$a_{\text{int}(\tau)} = 1 - \text{frac}(\tau) \quad (3.55)$$

$$a_{\text{int}(\tau)-1} = \text{frac}(\tau) \quad (3.56)$$

$$a_{\text{int}(\tau)} + a_{\text{int}(\tau)-1} = 1 \quad (3.57)$$

注意到,在本质上,上式所计算的分数延迟量是相邻两个延迟后采样的线性加权。例如,若所需分数延迟量为 6.5,则权重 $a_{\text{int}(\tau)}$ 和 $a_{\text{int}(\tau)-1}$ 等于 0.5,其整数延迟分别等于 6 和 7,有:

$$y_{FD}[n - \tau] \big|_{\tau=6.5} = 0.5 \cdot y[n - 6] + 0.5 \cdot y[n - 7]$$

而如果延迟量为 6.8 个采样点而不是 6.5,则分数延迟为:

$$y_{FD}[n - \tau] \big|_{\tau=6.8} = 0.2 \cdot y[n - 6] + 0.8 \cdot y[n - 7]$$

LTI 系统的好处是可在整数多路延迟模块中加入上述分数延迟差分方程,以获得一个动态可调的分数延迟滤波器,如图 3.29 所示。使用这一结构,多路延时模块可分为动态变化的整数延迟模块和分数延迟模块,式 3.54 的分数延迟部分可写成式 3.58 的差分方程,该式也提供了多路延迟滤波器中延迟器的整数部分。因此,整数延迟与分数延迟的严格组合可获得动态变化的分数延迟器。

$$y_{FD}[n - \tau] = a_0 \cdot y[n] + a_1 \cdot y[n - 1] \quad (3.58)$$

$$a_0 = 1 - \text{frac}(\tau) \quad (3.59)$$

$$a_1 = \text{frac}(\tau) \quad (3.60)$$

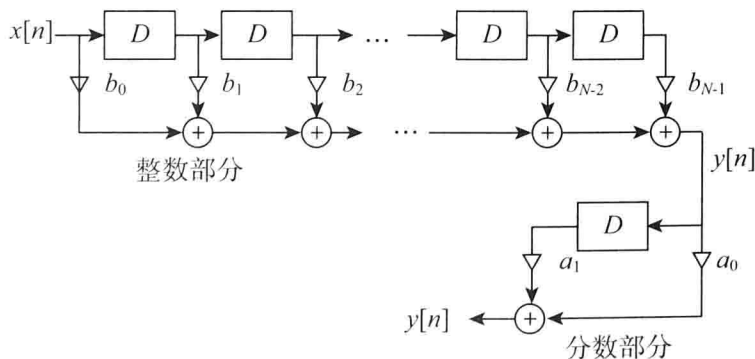


图 3.29 带有分数延迟的多路延迟滤波器

3.6.4 镶边效果

由第 3 章第 4.2.1 节可知,镶边效果与合声效果类似,也可利用可变延迟器实现。主要区别在于,使用附加声音的镶边效果器的延迟量变化范围在 1~10 毫秒,而合声效果的延迟量一般在 20~30 毫秒。与合声效果一样(而这里对于镶边器也一样),这种更短的时延并不会带来回声效果,而听起来更像是改变了音色。“镶边”(flanging)一词源于节目源(录制好的信号)相同的双轨磁带录音机同步回放的“卷轴到卷轴”(reel-to-reel)的时代。在这种情况下,用手交替轻压每个磁带机的法兰(即用手轻压磁带卷轴边缘造成轻微不同步后再同步松开磁带卷轴的边缘),每一轨磁带机将被迫发生轻微减速。镶边效果器的系统框图如图 3.30 所示。由图可知,它的基本结构与合声效果器一致。

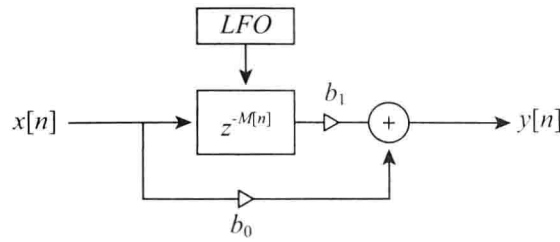


图 3.30 镶边滤波器结构

如上所述,镶边效果与合声效果的主要不同是延迟量的变化范围不同和所使用人声的数量。特别是,如式 3.61 的反馈前梳状滤波效果是极端镶边效果的特例——即一个输入信号 $x[n]$ (磁带机 1) 融合了它的延迟信号 $x[n-M]$ (磁带机 2)。

$$y[n] = b_0 \cdot x[n] + b_1 \cdot x[n - M] \tag{3.61}$$

由第 3.3 节可知,反馈前梳状滤波器通过调整零点塑造频率响应而产生梳状滤波。这正是原始信号及其微延迟信号沿着动态变化的延迟链叠加时发生的现象。这两种输入信号叠加的输出,可看成是输入信号变化延迟的相长或相消干涉产生的输出序列及其导致的陷波效果(零点位置)在频率响应上的映射图像。与合声系统相似,如第 3.6.3 节所述,镶边效果器也可用分数延迟器对延迟范围进行平滑扫描。

3.6.5 全通滤波器

顾名思义,全通滤波器让所有频率通过而不改变幅值。即它具有如图 3.31 所示的平直频率响应。

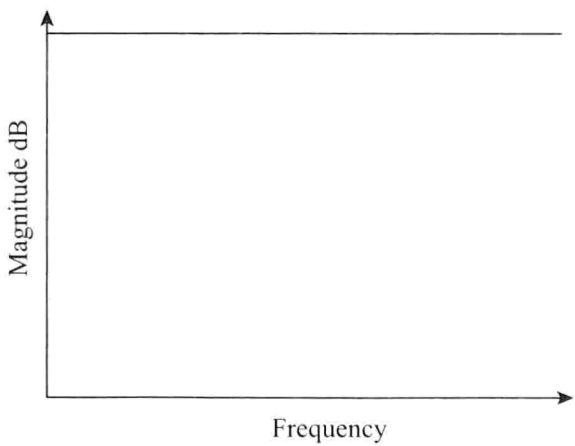


图 3.31 全通滤波器的频率响应

不过像这样的滤波器既没有多大意义,也根本引不起读者的兴趣,因为幅值响应没有变化,似乎信号本身也没有变化。这种观点在某种程度上是正确的,但也不完全准确——虽然信号幅度响应没有变化,但信号却可以发生改变。读者也许注意到本章处理滤波器时没有太多地提到相位。原因之一是在滤波和信号处理中,相位并不是最值得注意的显著特征——特别是相对而言,幅值和相位变化的感知几乎难以察觉。而在了解本节和下一节所介绍的全通滤波器后,希望读者会对相位及其在信号处理中的意义和重要性有更好的理解。

一阶全通滤波的差分方程如式 3.62,其 z 变换如式 3.63,系统框图如图 3.32:

$$y[n] = \alpha \cdot x[n] + x[n-1] - \alpha \cdot y[n-1] \quad (3.62)$$

$$H(z) = \frac{\alpha + z^{-1}}{1 + \alpha \cdot z^{-1}} \quad (3.63)$$

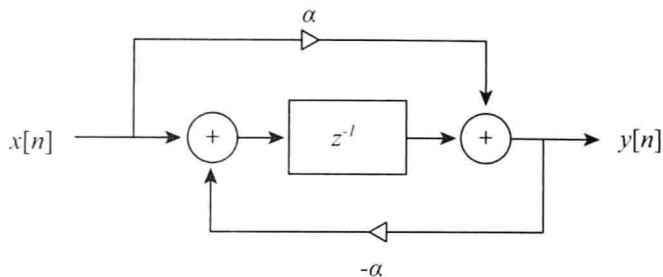
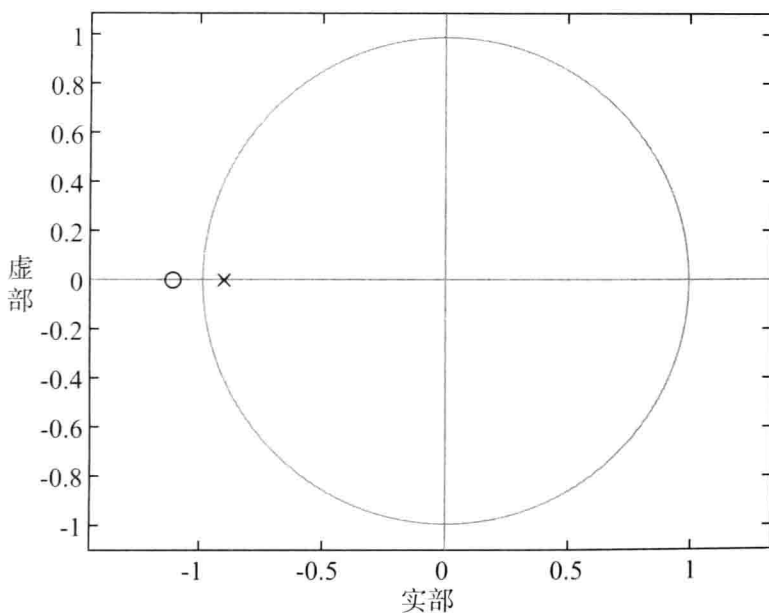


图 3.32 一阶全通滤波器的框图

图 3.33 进一步给出了系统零极点的分布,由图可知,零极点互相抵消(系统的推挽特征位于同一频点),并因此产生了平直的频率响应。

图 3.33 一阶全通滤波器的零极点, $\alpha=0.9$

为了证明传递函数的分子和分母确实相同,接下来计算传递函数的幅值响应 $|H(e^{j\theta})|$ (相当于单位增益响应),将式 3.63 的分子和分母乘以 $e^{j\theta/2}$ ——即将分子与分母乘以 $e^{j\theta/2}$ 而不改变其他参数,类似于将该分式乘以一个因子。由乘积结果可得出我们将要寻找的答案:

$$\begin{aligned} H(e^{j\theta}) &= \frac{\alpha + e^{-j\theta}}{1 + \alpha \cdot e^{-j\theta}} = \frac{e^{j\theta/2}}{e^{j\theta/2}} \cdot \frac{\alpha + e^{-j\theta}}{1 + \alpha \cdot e^{-j\theta}} \\ &= \frac{\alpha \cdot e^{j\theta/2} + e^{-j\theta/2}}{e^{j\theta/2} + \alpha \cdot e^{-j\theta/2}} \\ &= \frac{\alpha \cdot \{ \cos(\theta/2) + j\sin(\theta/2) \} + \{ \cos(\theta/2) - j\sin(\theta/2) \}}{\{ \cos(\theta/2) + j\sin(\theta/2) \} + \alpha \cdot \{ \cos(\theta/2) - j\sin(\theta/2) \}} \\ &= \frac{(1 + \alpha) \cdot \cos(\theta/2) + j(\alpha - 1) \cdot \sin(\theta/2)}{(1 + \alpha) \cdot \cos(\theta/2) - j(\alpha - 1) \cdot \sin(\theta/2)} \end{aligned}$$

$$\begin{aligned} &= \frac{r \cdot e^{+j\theta/2}}{r \cdot e^{-j\theta/2}} \\ &= \frac{e^{+j\theta/2}}{e^{-j\theta/2}} = 1 \cdot e^{+j\theta} \end{aligned} \tag{3.64}$$

这一计算技巧使分子和分母形成了一个共轭对,便于计算幅值频率响应 $|H(e^{j\theta})|$ 。为了简化,在计算式 3.64 的中间过程中,令 r 等于由实部和虚部定义的欧氏距离(通常,欧式距离是实部与虚部平方和的平方根),如 3.65。

$$r = \sqrt{\{(1 + \alpha) \cdot \cos(\theta/2)\}^2 + \{(\alpha - 1) \cdot \sin(\theta/2)\}^2} \tag{3.65}$$

由上式可知,分子和分母形成共轭对时,幅值响应为单位增益即 $|H(e^{j\theta})| = 1$ 。接下来,跳至讨论全通滤波器的真正议题,相位问题。计算虚部和实部之比的反正切函数,由 α 表示的相位,如式 3.66。图 3.34 与图 3.35 给出了该相位响应, α 值的步进增量为 0.1。

$$\begin{aligned} \phi(\theta) &= \tan^{-1} \left[\frac{(\alpha - 1) \cdot \sin(\theta/2)}{(\alpha + 1) \cdot \cos(\theta/2)} \right] - \tan^{-1} \left[- \frac{(\alpha - 1) \cdot \sin(\theta/2)}{(\alpha + 1) \cdot \cos(\theta/2)} \right] \\ &= \tan^{-1} \left[\frac{(\alpha - 1)}{(\alpha + 1)} \cdot \tan(\theta/2) \right] + \tan^{-1} \left[\frac{(\alpha - 1)}{(\alpha + 1)} \cdot \tan(\theta/2) \right] \\ &= 2 \cdot \tan^{-1} \left[\frac{(\alpha - 1)}{(\alpha + 1)} \cdot \tan(\theta/2) \right] \end{aligned} \tag{3.66}$$

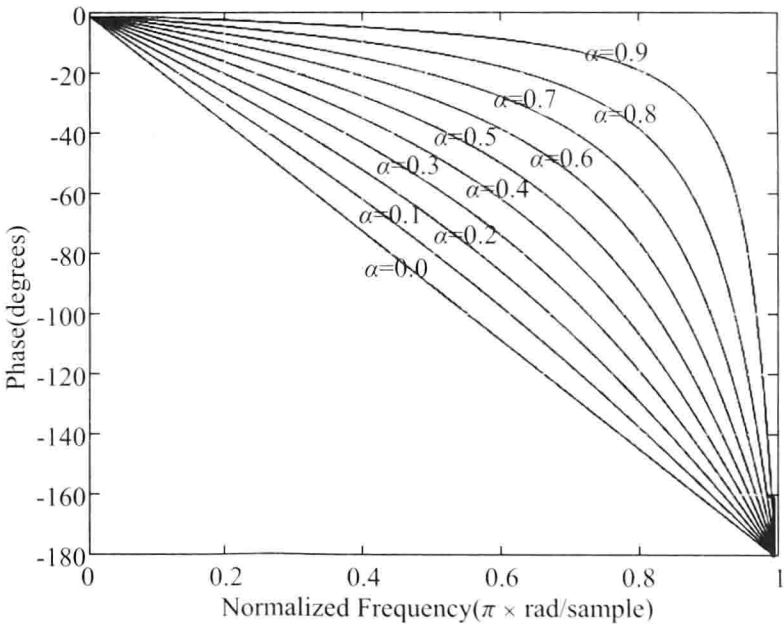


图 3.34 全通滤波器的相位响应(单位:角度)

由上述分析可知,滤波器系数 α 对图 3.34 与图 3.35 的相位响应的影响——滤波器系数 α 的值越大,输出结果信号的分数延迟量越低,且相位响应在低频至中频几乎是线性的(DC 分量至 $\pi/2$ 附近的频段),频率接近奈奎斯特频率时非线性特征变得越来越明显。因此,全通滤波器常通过选择适当的滤波器系数 α 的值来设计分数延迟器,它对于不包含高频的信号特别有效。

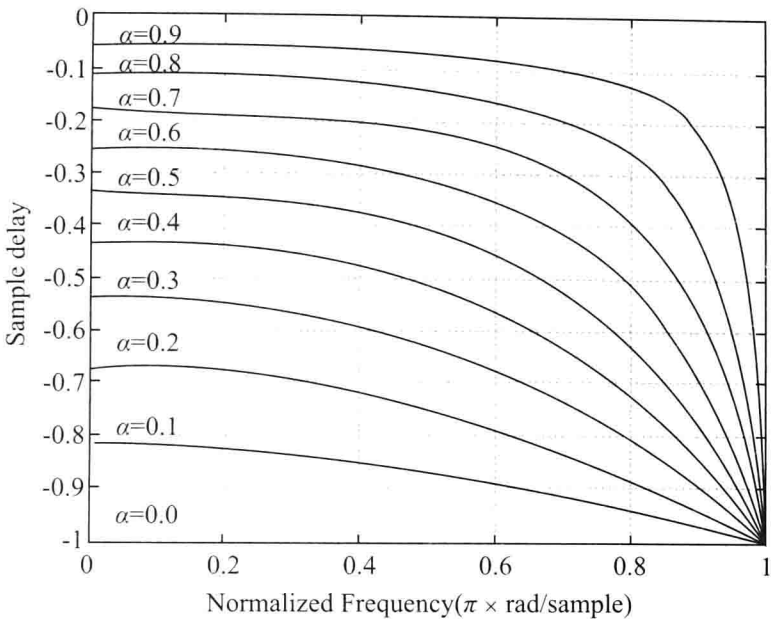


图 3.35 全通滤波器的相位响应(单位:分数采样延迟)

3.6.6 基本全通滤波混响器

混响是声音传播和声音感知中非常有趣而且重要的部分。无论何时何地,听到的声音或多或少都包含了来自声波反射界面或物体的反射声。读者也许体验过一些长混响的空间,如教堂、地铁站、银行和家里装了对声音有很好的反射作用的硬质表面浴室(笔者花了约一个星期给家里的浴室安装了大理石地砖——虽然以后不会再这样做了——但得到的经验是,浴室的声学效果变化显著)。如图 3.36 所示,混响通常来自于听众对声波传播路径中经墙壁、天花板或地板,以及任何其他表面和障碍的反射声的累积效应。有些反射声需要较长时间才能听到,是因为它们由较远的墙壁反射或经过多次反射后才回到听者的位置,有些反射声则会很快返回到听者的位置。显然,声波传播距离越远,或经反射的表面越多,则相对于迅速返回听者位置的声音来说,能量损失越大。这是合成人工混响的重要依据。

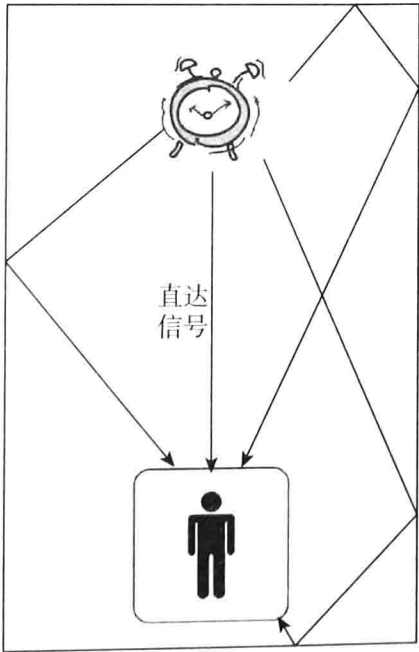


图 3.36 早期反射声

较早返回至听者的反射声称为早期反射声(early reflection),由声源直接传播至听者的声音称为直达声(direct sound)。早期反射声的时延约在 100 毫秒以下,经头几次反射而返回听者的声音有利于听者判断空间或房间的大小。早期反射声通常分为一阶和二阶反射声,对应于在返回听者之前声波经过的反射界面的数量。经过几次反射的反射声融合在一起就是通常所说的混响(reverberation)。图 3.37 给出一个脉冲信号的混响特征,其中 x 轴为时延, y 轴为反射后脉冲的幅值。由图 3.37 可知,声音反射图的幅值逐渐降低,显示了随时间衰减的振幅包络。这一现象很容易理解,反射面距离脉冲声源较远或反射界面较多时,脉冲信号将需要经过更长的传播距离才能返回听者,对听者来说,脉冲信号在传播路程中的能量损失也越大。同时也可注意到,随着时延的增长(一般在 100 毫秒左右),这些反射后的脉冲声融合成无数早期反射声,在本质上形成了混响效果。

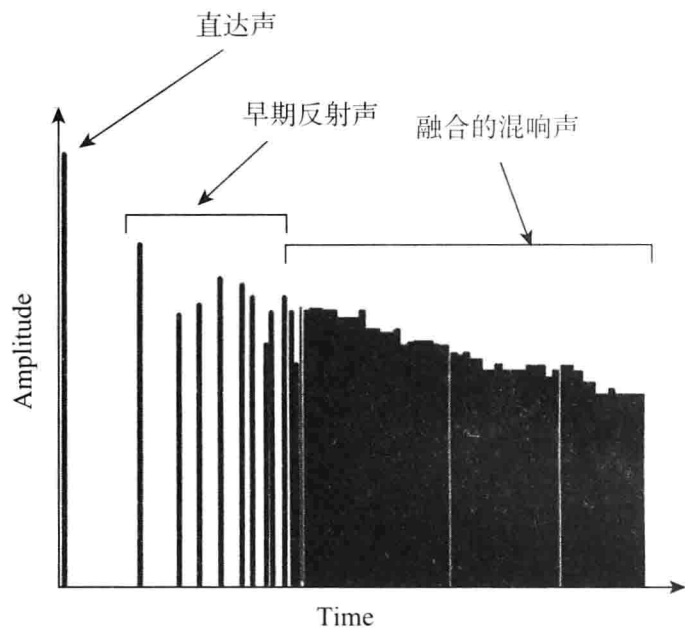


图 3.37 单个脉冲信号的混响特征

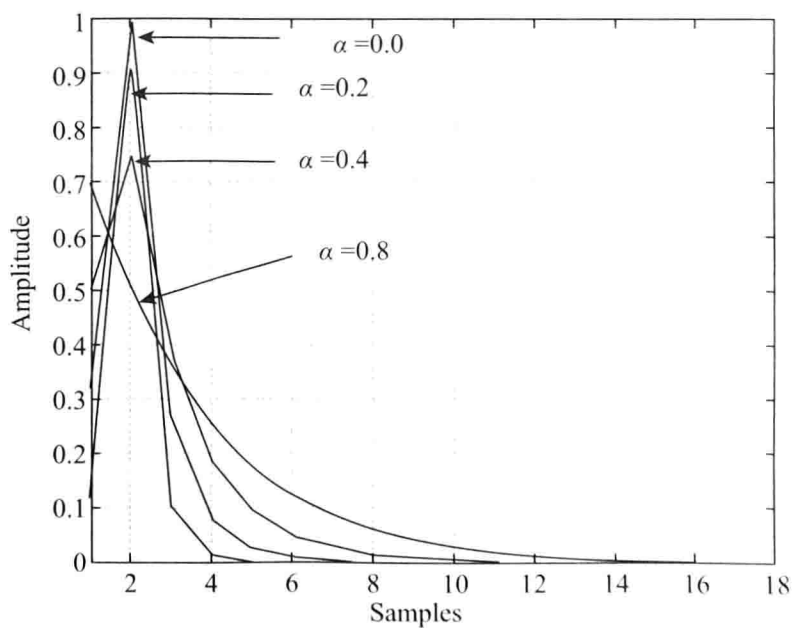


图 3.38 全通滤波器的脉冲响应

接下来回到式 3.62 的全通滤波器,观察其脉冲响应如图 3.38。由图可知两个主要结论:一是脉冲响应输出的衰减平滑,二是衰减特征由滤波器系数 α 控制(脉冲响应的绝对值更清晰地表现了该衰减特征)。

替换式 3.62 的初始全通滤波器,并增加延迟长度以使反馈与反馈前分量具有更大的延迟 N ,如式 3.67 所示:

$$y[n] = \alpha \cdot x[n] + x[n - N] - \alpha \cdot y[n - N] \tag{3.67}$$

绘制式 3.67 的脉冲响应可知:脉冲响应具有 N 个采样延迟,同时保持了原全通滤波器的衰减特征,如图 3.39、图 3.40 和图 3.41。

因此,高于一阶的全通滤波器可用于模仿声音通过界面的反射特征,并用于流行的非卷积人工混响算法中。还需要注意的是,脉冲响应随着时间序列的增长而衰减,就好像被或远或近的墙面所反射一样。

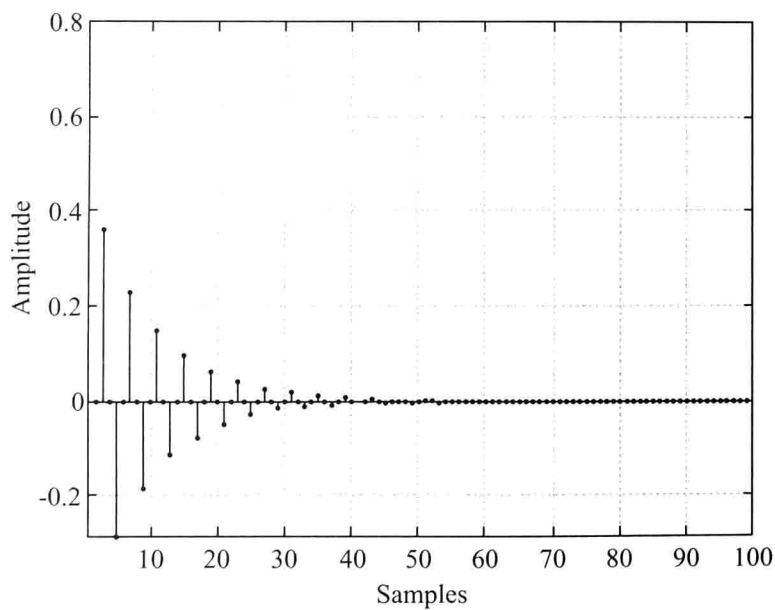


图 3.39 原始全通滤波器, $N=1, \alpha=0.8$

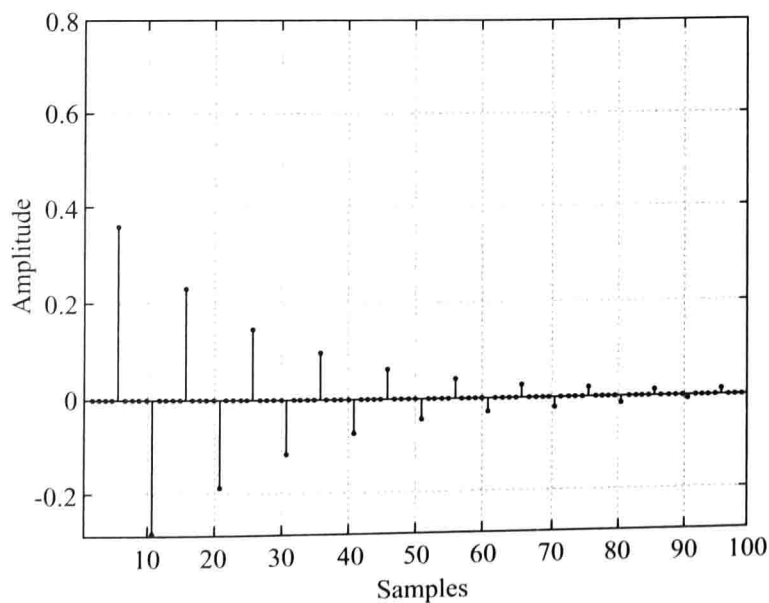
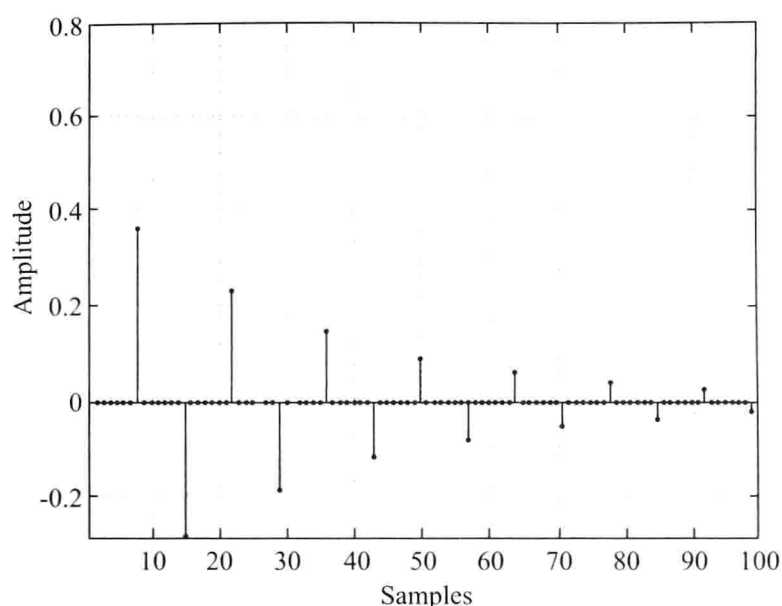


图 3.40 全通滤波器, $N=4, \alpha=0.8$

图 3.41 全通滤波器, $N=6, \alpha=0.8$

以下是一个基本全通滤波混响器算法的 MATLAB® 代码。

```
a = 0.9; % the all pass coefficient
fs = 20000; % sampling frequency
ratio = 1/25; % 1/25th of a second for reflection of sound

% reflection length in samples: filter order
reflectionLength = round(fs*ratio);

% impulse response
x = rand(1, fs/20); % make short impulse signal 1/20th sec
x = x - mean(x); % get rid of DC
x = [x zeros(1, fs*2)]; % pad with zeros to make actual impulse

% filter signal via all-pass filter to make reverb
y = filter([a zeros(1, reflectionLength) 1], ...
          [1 zeros(1, reflectionLength) a], x);

soundsc(y, fs)
```

代码 3.10 基本的全通混响器

在该 MATLAB® 实例中,用全通滤波器实现人工混响时将反射参数设为 1/25 秒。代码 3.10 直接使用了之前代码实例中学到的所有函数与命令。在大多数情况下,这种全通混响器的效果听起来不如所预期的那么真实,须进一步优化。

曼弗雷德·施罗德(Schröder, Logan 1961; Schröder 1962)在人工混响方面做了大量的研究,并开始试验使用全通滤波器和多全通滤波器串联结构来获得更好的结果尽管他名字中的字母为 ö,但如果用“oe”这一拼法替代 ö 来搜索他,可在网络中得到更多的资料。如图 3.42 所示,他还在全通滤波器后加入了一组并行梳状滤波器,以获得比仅使用单个全通滤波器更好的人工混响效果。

自施罗德的早期研究开始,人工混响的研究获得了长足的发展(虽然从在混响室中重放录

音到对传声器录制的原始干声信号加入混响还有很长的路要走),包括在反馈回路中使用低通滤波器来增强高频分量的衰减,如弹拨乐器算法。其他创新、优化和附加的改进算法还包括对特定房间尺寸、刻意几何形状以及声源和听者的位置进行建模。

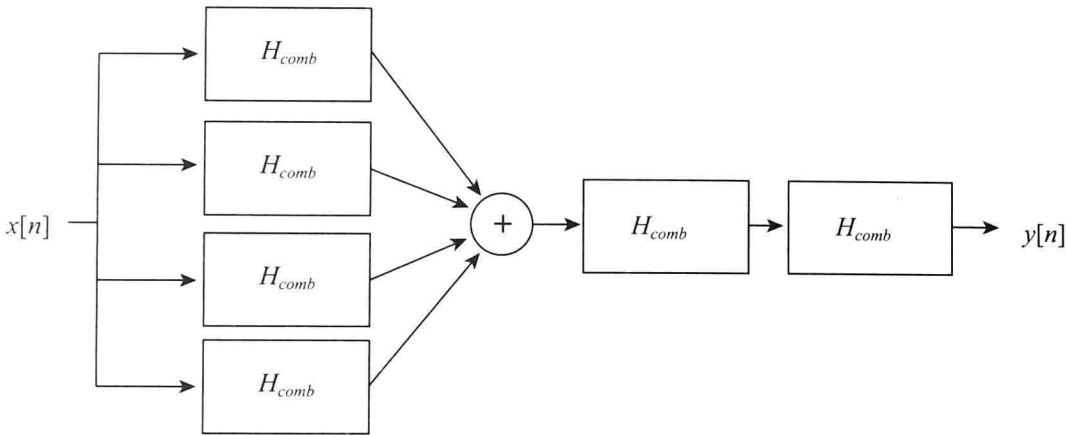


图 3.42 数字混响器的施罗德模型

4. 乐例分析

有许多有趣的滤波器音乐应用实例,可以毫不夸张地说,现今几乎所有录制和创作的音乐作品都使用了各种各样的滤波器。例如麦当娜(Madonna)歌曲《音乐》(Music,2000),其中人声部分使用了带通滤波器。又如吉他手吉米·亨德里克斯(Jimmy Hendrix)则使用了 Dunlop Cry Baby JH-1 型吉他“哇哇”声踏板。而一个非常有趣的大规模使用梳状滤波器的作品是保罗·兰斯基(Paul Lansky)的《夜间交通》(Night Traffic,1990)。这件作品的核心声音来自于新泽西州普林斯顿地区的一号公路,作曲家在这里录制了汽车在两车道高速公路上加速的声音。保罗·兰斯基将梳状滤波器的使用贯穿全曲,并由此创作了一首音乐性的作品,而不是为了演示梳状滤波器本身的特性。作曲家利用同时发声和原始声源(及其模糊化的声音)的概念,对这些声音进行了连接、组织与叠加处理,其产生的音响感觉已经超越了城市喧嚣声与汽车噪音本身的意义。兰斯基通过梳状滤波器对快速而乏味的交通噪声进行处理,使声源富于音乐性和戏剧,从而开辟了新的听觉世界,给听众带来了与截取汽车行驶的自然声不同的戏剧化音乐体验。从某种意义上说,将交通噪声的模式化和动机作为创作手段也许是一种陌生而时髦的方式,它放弃了声源原有的声学特征,但却以音乐的笔触记录了个人或长或短的经历和心路历程。也许从该作品的创作中学到的一课是以田野录音的方式来体验人生,这远比正确使用头戴耳机来聆听音乐更为重要。也许一开始时,保罗·兰斯基记录一号公路汽车加速的噪声只是因为错误地佩戴了耳机——将耳机的右声道佩戴到了左耳上,而将耳机的左声道佩戴到了右耳。当作曲家使用立体声麦克风进行录制的时候,他几乎走错了方向并进入超车道,因为来自于耳机回放的声音使他略有走神。

另一个与本章专题有关的,令人兴奋的作曲家是阿尔文·卢西尔(Alvin Lucier)。该作曲家也许以非常规的作曲手法而闻名,他的创作过程更像是在进行物理学实验而不是音乐创作。其中,具有里程碑意义的作品之一是《我坐在一个房间里》(I am Sitting in a Room,1969)。这件作品的内容其实是作曲家在录音室中对作曲过程的一段叙述,全文如下:

"I am sitting in a room different from the one you are in now. I am recording the sound of my speaking voice and I am going to play it back into the room again and again until the resonant frequencies of the room reinforce themselves so that any semblance of my speech with perhaps the exception of rhythm, is destroyed. What you will hear, then, are the natural resonant frequencies of the room articulated by speech. I regard this activity not so much as a demonstration of a physical fact, but more as a way to smooth out any irregularities my speech might have."

是什么令这首作品如此让人着迷呢?记录的声音为了反映出房间的共振结构,录制时用房间脉冲响应进行了迭代处理,随着声音中的谐波越来越多,作品的末尾变得越来越不知所云,只保留了原语音的基本模态——让作品越来越具有音乐性。因此,虽然这首曲子以熟悉的语言开始,但结束时声音已完全不同——由房间共振所产生的音响效果取代了原来的语言陈述。这种为了音乐创作而提取房间共振结构的技术有些类似于语音共振箱(talk box)。对于语音共振箱,源信号(通常是电吉他)输入至共振箱时,该共振箱就成为了按所需腔体共振结构对信号进行调制的人声腔体。语音共振箱有时可看成一个波导管,源信号(如吉他)经一小段软管的波导之后到达箱体的开口处,就像嘴贴近了人声麦克风。通过这种装置,源信号(吉他)投射到了如歌手嘴部的共振箱中,因此可改变该人声腔体的形状对源信号进行调制。调制后的声音经传声器拾取后再通过扬声器放大,就可以制造类似电吉他说话的效果。许多音乐家都曾使用这种效果进行创作或者演出,包括平克·弗洛伊德(Pink Floyd)的作品《猪》(*Pigs*, 1977),邦·乔维(Bon Jovi)的作品《与祈祷者的生活》(*Living on a Prayer*, 1986)和尼克·迪克科维斯基(Nick Dikovsky)使用女妖共振箱(*Banshee Talkbox*)所创作的弦乐四重奏作品《弦乐的通道》(*Tube Mouth Bow String*, 2007)。说到该共振箱的名字“女妖”(banshee),也许与数字信号处理本身并没有太大关系,因为作曲家亨利·科维尔(Henry Cowell)确实有一首名为《女妖》的钢琴独奏曲(1925)——该作品使用了传统钢琴演奏中不常见的直接扫、刮、乱弹和拨弦的技法而不是使用电子方式对钢琴声进行调制。正如作曲家和演奏者们所说:“条条大路通罗马。”

参考文献与扩展阅读

- [1] Frerking Marvin, E. 1994. *Digital Signal Processing of Communication Systems*. Van Nostrand Reinhold. New York, NY, USA.
- [2] Karplus, K. and Strong A. 1983. "Digital Synthesis of Plucked String and Drum Timbres", *Computer Music Journal*, 7(2), 43 - 55.
- [3] Jaffe, D. and Smith J. O. 1983. "Extensions of the Karplus-Strong Plucked String Algorithm", *Computer Music Journal*, 7(2), 56 - 69.
- [4] Schroeder, M. R. and Logan B. F. 1961. "Colorless Artificial Reverberation", *Journal of the Audio Engineering Society*, 9(3), 192.
- [5] Schroeder, M. R. and Logan B. F. 1961. "Natural-sounding artificial reverberation", *Journal of the Audio Engineering Society*, 10(3), 291 - 233.
- [6] Moore, F. R. 1990. *Elements of Computer Music*, Prentice Hall; Englewood Cliffs, NJ, USA.
- [7] Moorer, J. 1979. "About this Reverberation Business", *Computer Music Journal*, 3(2),

13–28.

[8] Morse, P. M. 1936. *Vibration and Sound*. Published by the American Institute of Physics for the Acoustical Society of America; Woodbury, New York USA, 1976 (1st ed., 1936; 2nd ed., 1948).

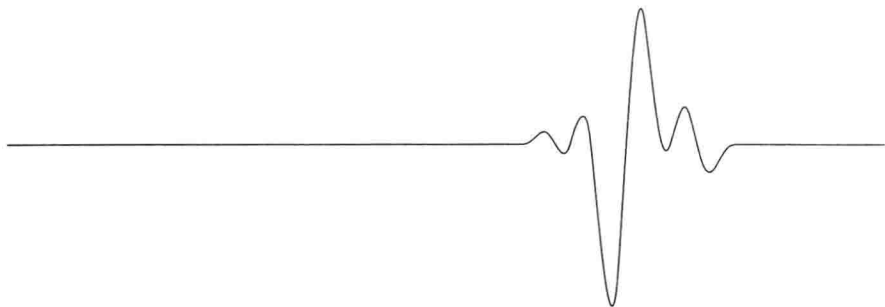
[9] Park, T. H. and Li Z. 2008. “All this Buzz about the Sitar: Physical Modelingly Speaking”, *Proceedings of the 14th International Congress on Sound and Vibration*, Daejon, South Korea.

[10] Porat, Bocz 1997. *A Course in Digital Signal Processing*. Wiley and Sons.

[11] Smith, J. O. 1992. “Physical Modeling Using Digital Waveguides”, *Computer Music Journal*, 16(4), 74–91.

[12] Sullivan, Chrlie R. 1990. “Extension the Karplus–Strong Algorithm to Synthesize Electric Guitar Timbres with Distortion and Feedback”, *Computer Music Journal*, 14(3), 26–37.

第 8 章 频域与傅立叶变换



1. 引言

至此,前面的章节主要集中讨论了时域信号处理的问题,同时对频域信号处理的相关概念也有提及。本章将完整讨论与频域有关的理论和内容。支撑频域理论的诸多概念将围绕一个极其重要的信号处理概念展开,它称为傅立叶变换。傅立叶变换也是数字信号处理(及其他研究领域,如数学、物理、电子工程和计算机科学)的强大工具之一,它广泛应用于声音的合成以及再合成、声音处理、滤波和谱分析等方面。虽然之前提到过许多关于声音对象的频率成分问题以及如何将声音信号分解成一系列特定正弦或余弦分量的问题,但还没有工具用于获得这些分量的特征。傅立叶变换恰恰就是最终获得和计算这些正弦分量的振幅、频率和相位的方法之一。

2. 加法合成

在介绍傅立叶变换这一难题之前,首先介绍“加法合成”(additive synthesis)及相关基本概念。这些初步的讨论能有助于更好地理解隐藏在傅立叶变换背后的思想。加法合成与第 7 章 7.2.1 节介绍的减法合成相反。减法合成用一个复杂信号(如方波或白噪声)合成具有特定频率且更单纯的信号。加法合成的基本方法不是强调特定的频率,而是恰恰相反——以特定正弦或余弦分量的叠加构建一个复杂信号。即通过具有确定振幅、频率或相位的特定正弦波或余弦波的叠加,来合成一个复杂信号。

诠释加法合成的典型范例是方波合成,其近似方程如式 2.1 所示,其中 f 为频率,单位为 Hz, k 为整数,用于确定每个正弦分量的频率($k=0$ 时对应着基频), n 为采样点序号。

$$y[n] = \sum_{k=0}^K \frac{\sin[2\pi \cdot f \cdot (2k+1) \cdot n]}{2k+1} \quad (2.1)$$

观察上式可知,频率分量都是基频的奇数倍。即生成的信号 $y[n]$ 是包含了基频及其奇次谐波的复合信号。此外,由于在构造谐波时以 $2k+1$ 为分母,则幅值随着谐波指数 k 的增加而降低。

图 2.1 至图 2.3 给出了方波加法合成算法的图像,基频 $f_0 = 1\text{ kHz}$,合成的方波分别带有 1、3、10、50、100 和 200 个谐波(K),每图的上图(每个图有两个子图)为用于合成复合信号并随谐波个数增加的单个正弦分量的波形(基频及其谐波),下图为通过这个正弦波合成的方波(x 轴为采样点, y 轴为振幅)。

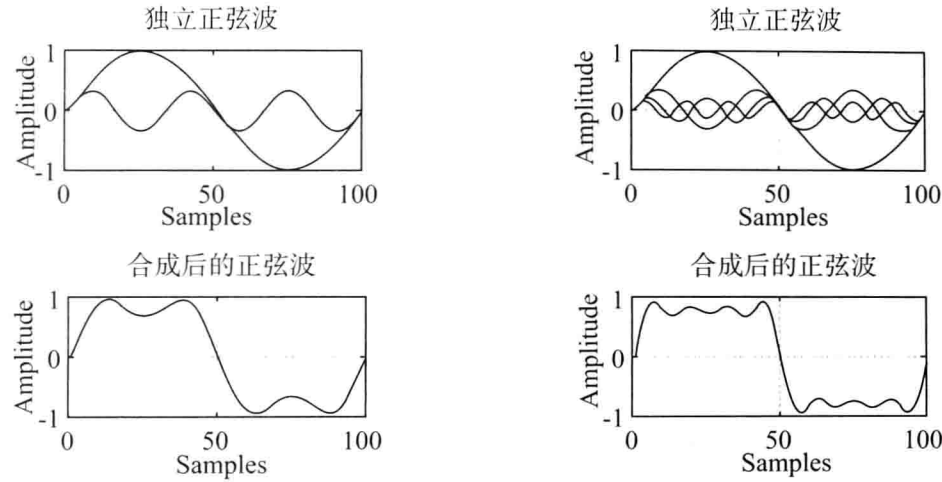


图 2.1 带有一个及三个谐波的方波加法合成

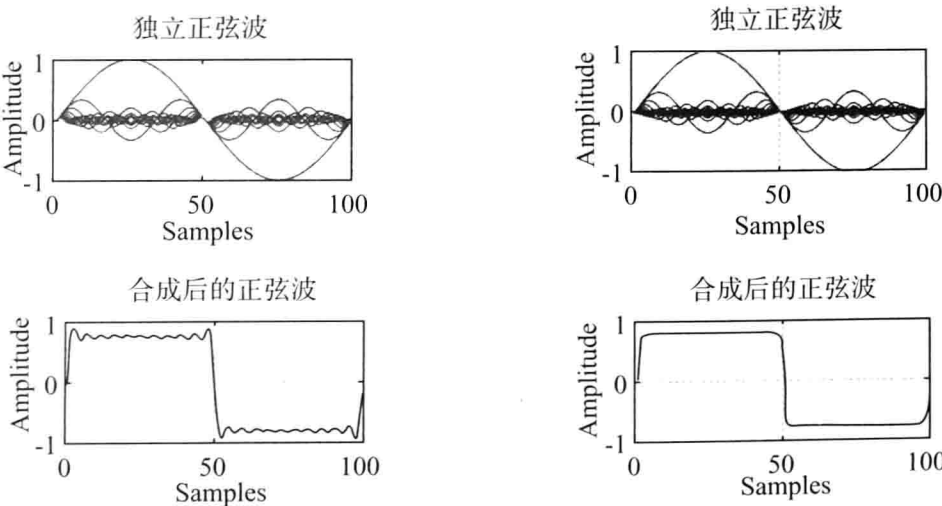


图 2.2 带有 10 个及 50 个谐波的方波加法合成

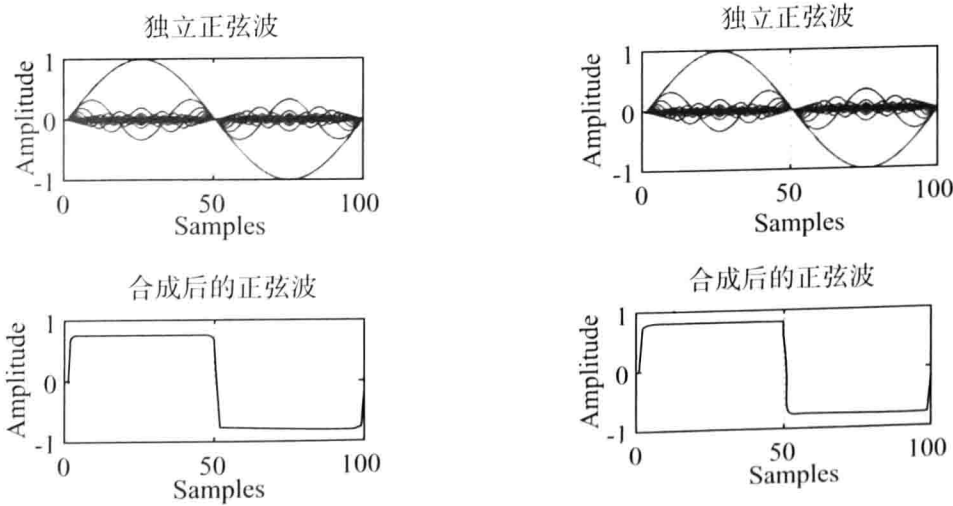


图 2.3 带有 100 个及 500 个谐波的方波加法合成

由图可知,随着叠加的正弦波越来越多(K 值越大),合成的波形就越来越接近方波。较长波长对应的低频分量决定了合成波形的一般形状——因为低频分量与高频分量相比振动更加缓慢。高频分量在给定时间帧内振荡越快(此例幅值较小),则其所调整合成输出的波形边缘越陡峭,方波特征越明显。

用正弦波合成三角波与合成方波相似,其谐波也是由奇次谐波组成,只不过谐波的幅值呈指数衰减,如式 2.2。

$$y[n] = \frac{8}{\pi^2} \sum_{k=0}^K \sin\left\{\frac{\pi \cdot (2k+1)}{2}\right\} \cdot \left\{\frac{\sin[2\pi \cdot f \cdot (2k+1) \cdot n]}{(2k+1)^2}\right\} \quad (2.2)$$

另一种称为锯齿波的复合波也可用同样的方法得到,如式 2.3,其中该例包含了完全的谐波序列(奇次谐波和偶次谐波)。与方波合成一样,大多数情况下,用于合成的正弦波越多,波形越接近于锯齿波。

$$y[n] = \frac{2}{\pi} \cdot \sum_{k=1}^K \sin\left(\frac{\pi \cdot k}{2}\right) \cdot \left[\frac{\sin(2\pi \cdot f \cdot k \cdot n)}{k}\right] \quad (2.3)$$

足以令人感兴趣的是,加法合成不仅可用于熟知的电子合成器,也可用于数字合成,就像一个机器由上万条电缆连接起来一样,在数字系统中多个信号元素也可以组成一个数据轮。流行的哈蒙德风琴(Hammond Organ)就是这样一种加法合成器,它利用可以放置 91 张盘片的“轮盘”(tonewheels),通过 9 个拉杆来控制轮盘所生成的正弦波的幅值。那么所有上述这些内容与以傅立叶变换有何关系呢?继续读下去就可找到答案。

3. 傅立叶变换

“傅立叶变换”(Fourier transform)是为纪念法国工程师让·巴蒂斯特·约瑟夫·傅立叶(Jean Baptiste Joseph Fourier)在该理论上的卓越贡献而命名的。简而言之,傅立叶变换是在 19 世纪初因求解数学方程的需要而发展起来的,它可将信号分解成无穷个具有既定频率和相位的加权正弦信号。如在加法合成与模拟方波的例子中,分解而得的权值与频率分量是一系列随阶数衰减幅值和奇次谐波。

时间频率连续的傅立叶变换(continuous-time and continuous-frequency Fourier transform)如式 3.1, $X^F(\omega)$ 是傅立叶变换结果, ω 为角频率,单位为 rad/s, $x(t)$ 为时间连续的输入信号。

$$X^F(\omega) = \int_{-\infty}^{+\infty} x(t) \cdot e^{-j\omega t} dt, \quad \omega \in R \quad (3.1)$$

上述傅立叶变换的定义只在积分有界时存在,即傅立叶变换存在的充分条件如下:

$$\int_{-\infty}^{+\infty} |x(t)| dt < \infty \quad (3.2)$$

由上述条件可知,就像使 IIR 滤波器保持稳定一样,傅立叶变换要求输入信号的能量时域积分有界。为了从变换后的频域信号得到时域信号,可使用逆傅立叶变换,定义如 3.3 所示:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X^F(\omega) \cdot e^{+j\omega t} d\omega, t \in R \quad (3.3)$$

上式是第2节由数字方波加法合成器的连续形式——即通过具有特定频率和相位的正弦波合成所需的时域信号(方波)。反之,对方波信号进行傅立叶变换,在考虑权重的情况下,可得到该方波的基频、谐波结构和相位值——该方波的特征结构由奇次谐波与按一定规律衰减的振幅构成,如式2.1所示。

4. 离散时域傅立叶变换与离散傅立叶变换

对于计算机应用软件,用积分而不是求和来实现式3.1的时间连续且频率连续的傅立叶变换当然是不现实的。因此,需要定义时间离散与频率离散的傅立叶变换形式。对输入信号进行采样并将时域积分转换为式4.1的求和形式、将角频率 ω 可转换为式4.2的数字频率 $\theta \cdot n$ 时,该离散信号的傅立叶变换为时间离散傅立叶变换 DTFT(discrete-time Fourier transform),由式4.3计算。获得完全离散的傅立叶变换形式的最后一步是对频率轴进行采样,并由此得到离散傅立叶变换 DFT(discrete Fourier transform)。稍后将讨论如何实现这最后一步。而现在回到时间离散傅立叶变换。只要满足式4.4与时间连续傅立叶变换类似的求和有界条件,时间离散傅立叶变换就存在。

$$\int_{-\infty}^{+\infty} \longrightarrow \sum_{n=-\infty}^{n=+\infty} \quad (4.1)$$

$$e^{j\omega} \rightarrow e^{j\theta n} \quad (4.2)$$

$$X^f(\theta) = \sum_{n=-\infty}^{n=+\infty} x[n] \cdot e^{-j\theta n} \quad (4.3)$$

$$\sum_{-\infty}^{+\infty} |x[n]| < \infty \quad (4.4)$$

DTFT 的逆变换如式4.5:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{+\pi} X^f(\theta) \cdot e^{+j\theta n} d\theta, n \in Z \quad (4.5)$$

需要注意的是,DTFT 的角频率还不是离散的,它可随时转换成真实频率(与 DFT 有关的离散频率将在稍后讨论)。DTFT 使用前面章节讨论过的单位为弧度/采样(radians/samples)的数字频率 θ 表示。考虑到数字形式,“数字频率”一词一开始会对连续数字频率 $\theta = \omega \cdot T$ 有些干扰,并自觉地认为它是离散的。不过,数字频率实际上指某个离散时刻 $n \cdot T$ 所对应的频率,与时间连续频率并不相同,离散频率将在介绍 DFT 时讨论。

DTFT 作用于经采样并经宽度为 N 的时窗截取的输入信号。对时间连续信号 $x(t)$ 在时域上进行采样读者已经非常熟悉了,而对于频率轴上的采样则较为陌生。为了理解在频率轴上的采样,首先需要掌握的是傅立叶变换的思想和时窗经傅立叶变换后由时域谱到频域谱的映射变换。因此,在介绍 DFT 和频率采样的相关细节之前,先观察一个经时窗处理后的时域帧转换为频域帧的傅立叶变换特征。图4.1显示了以离散信号 $x[n]$ 为输入信号的离散傅立叶转换为频谱 $X^d[k]$ 的过程,其中 k 为离散频率指数。

图4.2为由三次谐波用加法合成的基频为1Hz,采样率为44.1kHz的时域离散信号 $x[n]$

的前 1,536 个采样信号。按离散傅立叶变换 DFT 的过程,首先需要用时窗截取时域信号,如图 4.3。此例采用了一个长度 N 为 512 个采样点的矩形窗,将输入信号 $x[n]$ 分成帧长度均为 512 个采样点的三个长度相等的部分。图 4.3 的中图显示的是第一个时窗信号。

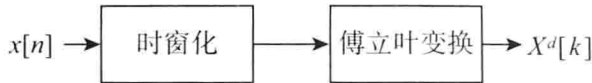


图 4.1 傅立叶变换的过程

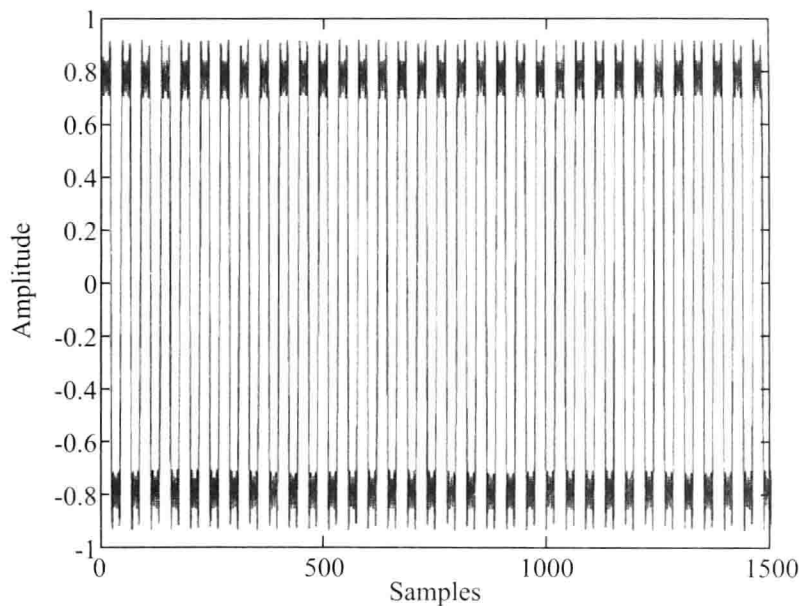


图 4.2 带有三次谐波的方波的前 1,536 个采样

图 4.3 的下图是时域中前 512 个采样点的傅立叶变换结果,需要强调的是,这一变换是 512 个时域采样到 512 个频域采样的映射。换句话说,由时域转换至频域时,原来在 x 轴上表示时刻的采样点序号 n 变成了在频域上的频率指数 k 。被时窗截取的前 512 个时域采样,在傅立叶变换后被 512 个频率采样合为了 1 帧 (frame),每帧包含的 512 个频率采样称为一个频谱。此时 x 轴 (频率轴) 上的离散点通常被称为频率帧 (frequency bin) 或帧 (bin)。

为了更清楚地理解傅立叶变换的过程,接下来以使用红、绿、蓝色度为参数的 RGB 色彩模式为例进行类比分析。计算机系统中,8 位 RGB 系统的每个颜色参数通常是用 0 ~ 255 的整数来表示。也就是说,如果这三种基色的组合和权重得当,就可以获得所需要的颜色。现在考察一部每秒 24 帧的标准电影,它投射在银幕上时会产生连续运动的视觉错觉,很像采样后的声信号序列 [如采样率为 441,000 个采样点/秒 (samples/sec)] 会产生连续声的听觉错觉。再回到 RGB 系统,如果要分析影片片断的色彩映射——以 1 次 6 帧为单位。这一时窗大小在 24 帧系统中相当于 1/4 秒 (如果采用 24 帧/秒的标准),用 RGB 所表示的是以 6 帧为一个整体的由红、绿、蓝整数色度级所表示的色彩信息。在这一分析系统中,得到的是整部影片中每四分之一秒的 RGB 色彩信息。从本质上看,就将影片由运动画面组成的时域转换到了“基色域” (primary-color-domain),该色域仅由每一组所分析帧的 RGB 级表示 (当然可以把更多或更少的帧作为一个分析单位)。因此, x 轴也从时域转换到了由 RGB 系统表示的色域结构,如图 4.4。

离散傅立叶变换在一定程度上可与上例进行类比,其分析帧长为 512 个采样点,而不是影片中的 6 个图像帧,每帧再用 512 个采样点代替权重不同的 RGB 三基色,并使用了 512 个具有

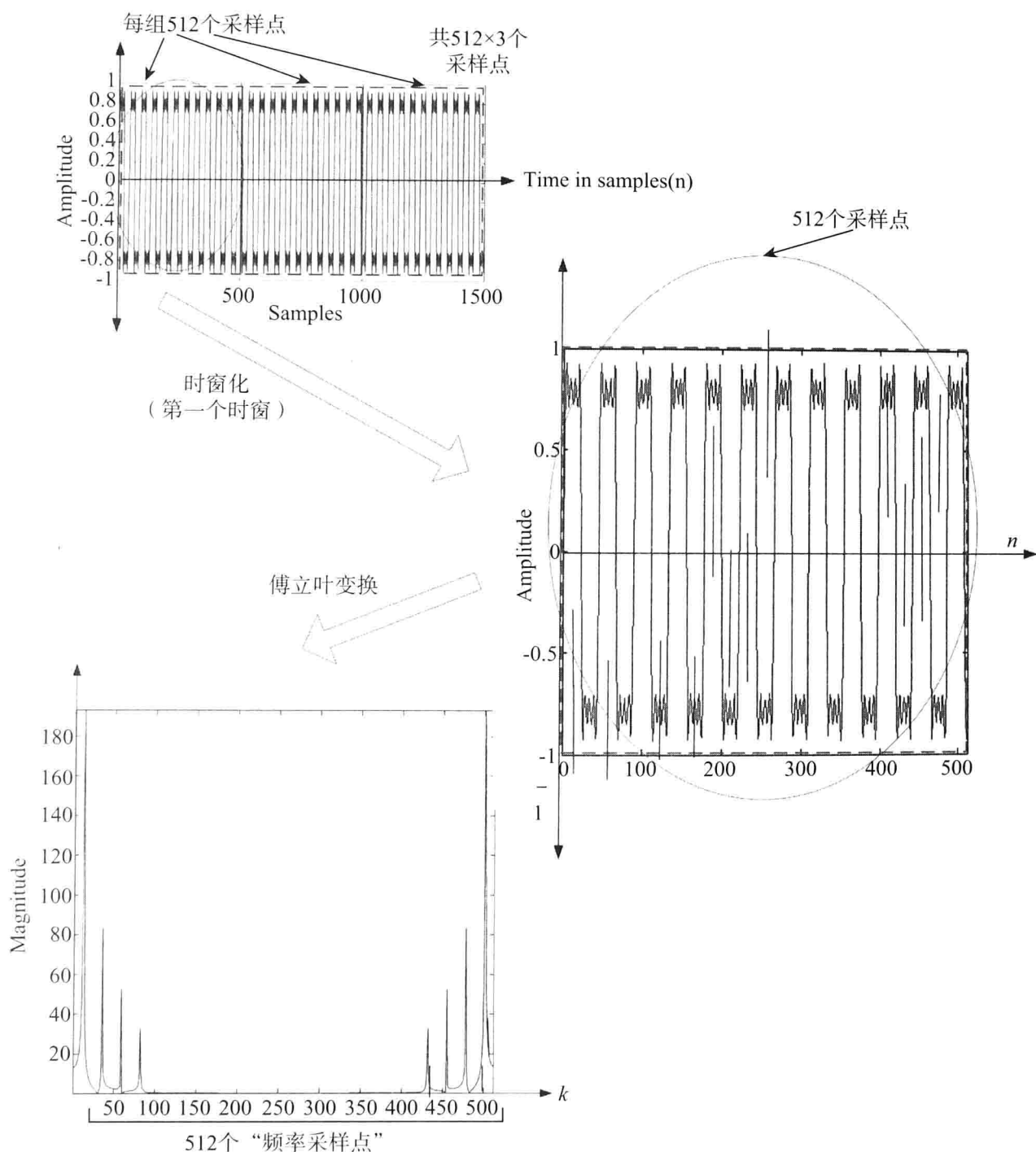


图 4.3 时域到频域的变换

各自幅值和相位的频率分量。每个频率分量表示一个具有特定频率和各自幅值的正弦分量,表征了时窗化后包含 512 采样点的信号的频域特征。因此,这一包含 512 个采样点的时窗化信号成为了由 512 个具有自频率、振幅和相位的正弦分量重构的频域信号。总而言之,从正弦分量叠加的角度看,512 个频域的正弦分量为 512 个时域采样提供了一个概览。

现在回到基频构建于前 3 个奇次谐波的方波加法合成的例子,其傅立叶变换如图 4.5,由图可知该频谱有 8 个正弦分量(显著峰值)而不是 4 个——是实际正弦波数量的两倍。这对我们期待得到一个基频和三个奇次谐波而言有些出入。仔细观察发现,前 4 个谱峰和后 4 个谱峰似乎存在镜像对称关系。稍后将讨论发生这一现象的原因,而引入欧拉公式(由正弦和余弦分量

组成)应该不难解释这一现象(与频率响应分析类似)。如果这是频谱得到的对称重复分量,就表明全部 512 个频率帧中只有一半有意义。因此,正如所预期的那样,如果考察前 256 个频点,能明显观察到基频和幅值衰减的前 3 次奇次谐波,如图 4.6:

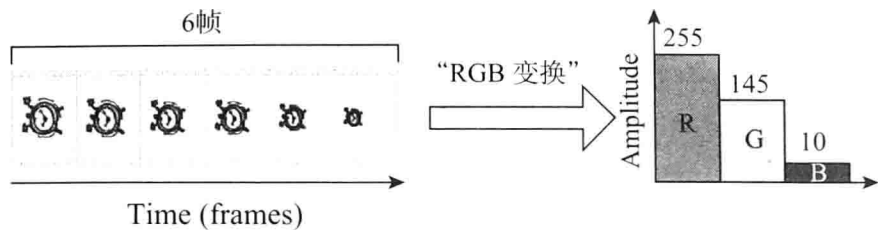


图 4.4 彩色卡通片前 6 帧的 RGB 级

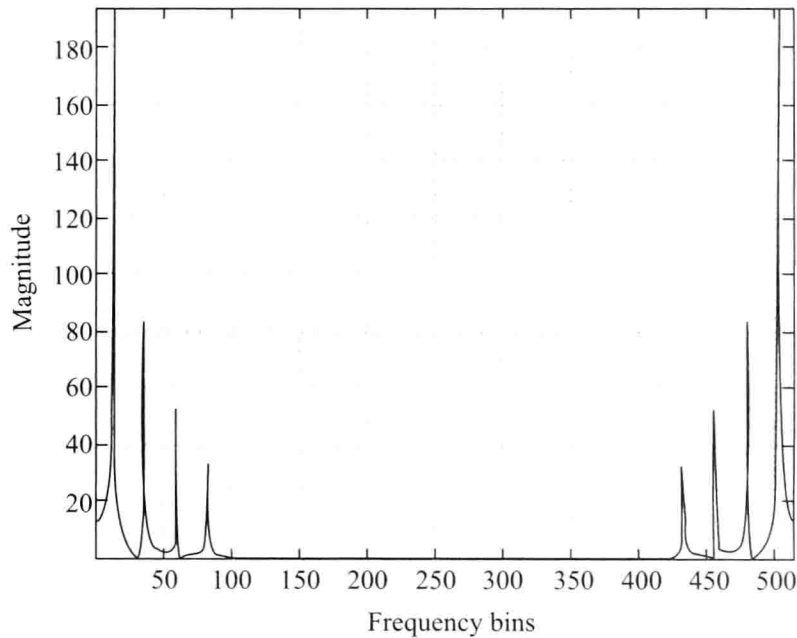


图 4.5 方波的傅立叶变换

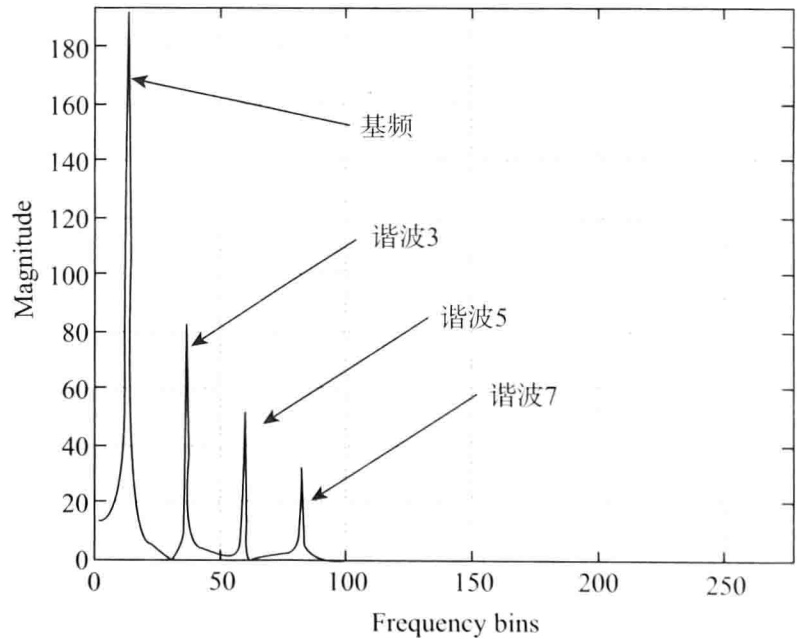


图 4.6 基频及其三次谐波的频窗

接下来首先通过时间离散傅立叶变换 DTFT 来分析 DFT 背后的数学意义, n 为采样点序号, θ 是单位为弧度/采样的数字角频率。

$$X^f(\theta) = \sum_{n=-\infty}^{n=+\infty} x[n] \cdot e^{-j\theta n} \quad (4.6)$$

其中

$$e^{-j\theta n} = \cos(\theta \cdot n) - j\sin(\theta \cdot n) \quad (4.7)$$

对频率轴进行采样(与对时间轴进行采样相同), 范围由 0 至 2π , 相当于真实频率 0 至 f_s Hz, 式 4.8 将该频率范围等分为 N 个频段。

$$\text{单位频率} = \frac{2\pi}{N} \quad (4.8)$$

和时域采样一样对频域进行采样, 在时域采样中采样周期 T (单位为秒) 表示样本间的采样间隔, 此处以 $2\pi/N$ 作为频率采样间隔(单位为弧度)。因此, 采样后的数字频率 θ 可表示为离散数字频率 $\theta[k]$, k 为表示离散频率序号的整数(通常也指频率帧或帧)。如下式 4.9, N 为 DFT 时窗长度(前例 $N=512$)。

$$\theta[k] = \frac{2k\pi}{N}, 0 \leq k \leq N-1 \quad (4.9)$$

将式 4.6 中的 θ 用式 4.9 代入可得 DFT 的计算式, 如式 4.10, 其中, $X^d[k]$ 为离散频率帧或序号 k 所对应的 DFT 变换, $x[n]$ 为输入序列, 时域采样时窗长度为 N 。

$$X^d[k] = \sum_{n=0}^{n=N-1} x[n] \cdot e^{-j\frac{2\pi kn}{N}}, 0 \leq k \leq N-1 \quad (4.10)$$

同时可注意到将 k 的定义域代入式 4.9 可得离散数字频率 $\theta[k]$ 的定义域, 如式 4.11:

$$0 \leq \theta[k] \leq \frac{2k\pi(N-1)}{N} \quad (4.11)$$

即采样后频率 $\theta[k]$ 的定义域由 0 至 2π , 最小步进为单位频率间隔 $2\pi/N$ 。采样后频率不包括本身为 0 (DC 直流分量) 的第 N 阶频率分量, 因为频谱始终会包含该分量。而时窗化输入信号的时域采样个数总是等于频率帧(频率采样)的个数。

DFT 的逆变换 IDFT 如式 4.12, 为方便描述, 常引入符号 W_N 来表示 DFT 或 IDFT 的自然指数项, 如下:

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X^d[k] \cdot e^{+j\frac{2\pi kn}{N}} \quad (4.12)$$

$$W_N = e^{j\frac{2\pi}{N}} \quad (4.13)$$

采用该简写符号, DFT 也可写成式 4.14:

$$X^d[k] = \sum_{n=0}^{n=N-1} x[n] \cdot e^{-j\frac{2\pi kn}{N}} = \sum_{n=0}^{n=N-1} x[n] \cdot W_N^{-kn} \quad (4.14)$$

4.1 DFT 的幅值、相位与其他基本性质

由 DFT 的推导过程与欧拉公式可知,输出 $X^d[k]$ 将是一个复数——这里的 DFT 曲线是幅值曲线。由于 DFT 的输出是一个复数,可由式 4.15 所示向量表示,幅值常用于分析 DFT,对于帧 k ,幅值计算如式 4.16,相位定义反正切函数,如式 4.17 所示:

$$X^d[k] = a_k + jb_k \tag{4.15}$$

$$|X^d[k]| = \sqrt{a_k^2 + jb_k^2} \tag{4.16}$$

$$\angle X^d[k] = \arctan \frac{b_k}{a_k} \tag{4.17}$$

图 4.7 给出了由基频及其前几个奇次谐波加法合成方波的 32 点 DFT 变换结果。如上节所述 DFT 变换后的频谱以指数 $N/2$ 所对应频点呈现镜像对称,则此例由 DC 分量(0)至奈奎斯特频率($N/2 - 1$)所对应的频率序数 $k=0 \sim 15$ 。第 16 阶($k=15$)之后的分量与之前的分量呈镜像对称关系,DC 直流分量位于频率轴的最右端。

图 4.7 采用了三种不同单位来表示同一个频率轴,它们的含义如表 4.7。对于采样率为 $f_s = 44.1 \text{ kHz}$ 的系统,角频率 $\theta[15]$ 约为 2.94 弧度,对应的真实频率约为 20,671.875Hz。向右移动一个“点”的步进,频率将增加 π 弧度或 22,050Hz。

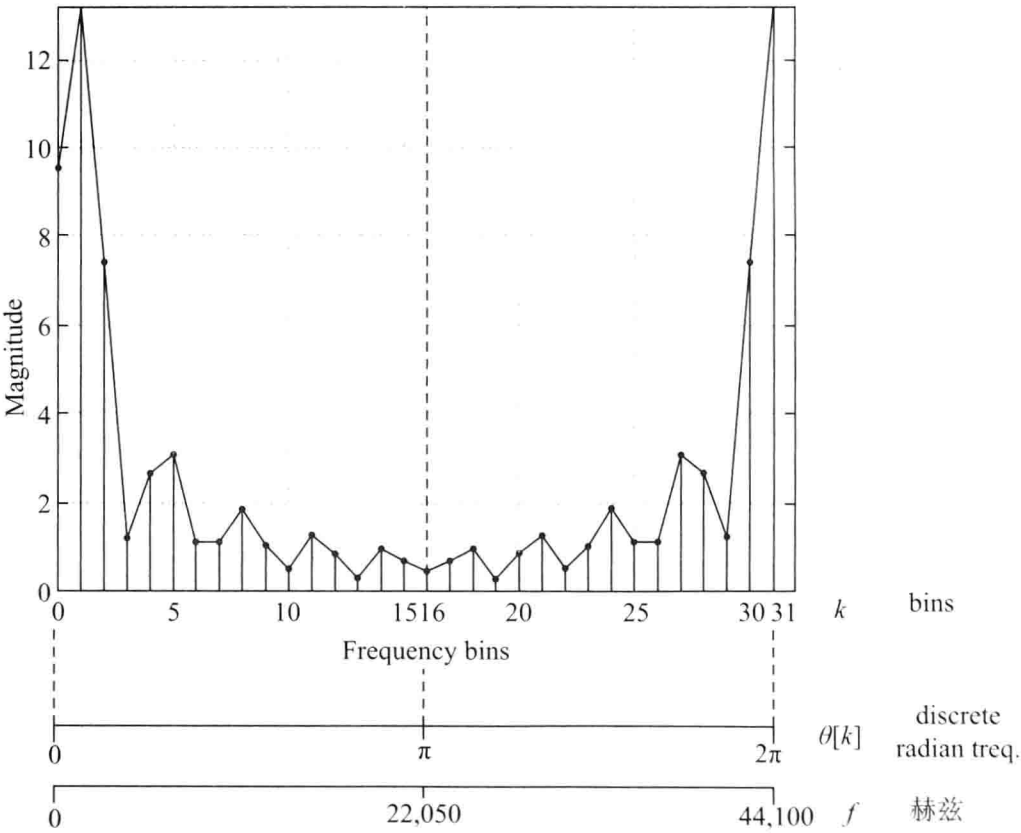


图 4.7 DFT 的对称特征, $N=32$, $f_s=44.1 \text{ kHz}$

表 4.1 频率轴的单位

单 位	描 述
频段 k (Frequency bin)	$k=0\cdots N-1$, N 是整数为 DFT 变换长度。
采样后的角频率 $\theta[k]$ (Sampled angular frequency)	$\theta[k]=2k\pi/N$ 。
真实频率(单位 Hz)	$f_{\text{Hz}}=f_s\cdot(k/N)$ 为采样率。

4.2 DFT 的时域分辨率与频域分辨率

在上节图 4.7 中提到使用前几次谐波进行合成,而该图给出的谐波数量与用于合成的谐波数量并不相符。乍一看,也许可认为图中只显示了 4 个谐波与 1 个基频,因为后 5 个峰与前 5 个峰是镜像对称的。但这种观点是不正确的。如果使用更小的频率采样间隔,也就是更大的 DFT 窗如 128 点,则对于同样信号得到的频谱完全不同,如图 4.8:

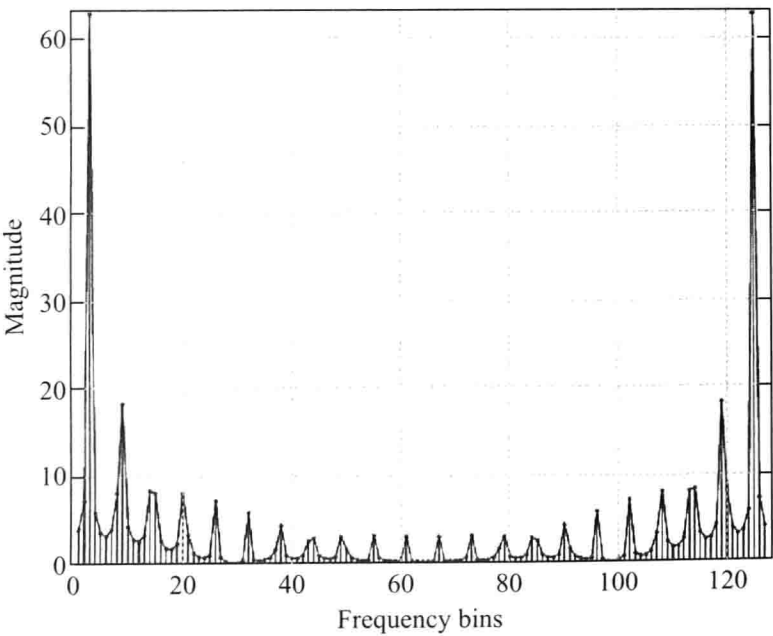


图 4.8 10 个谐波方波的 128 点 DFT 变换

随着 DFT 长度的增大,频率分辨率也随之增大,或反之,由式 4.9 定义的频率采样间隔降低了。 $N=32$ 增大至 $N=128$ 后,频率轴被等分为 128 个帧而不是 32 个(频率采样间隔为 $44100/128=344.53\text{Hz}$ 而不是 $44100/32=1378.125\text{Hz}$)。如果使 N 值达到更高的 1,024,则频率采样间隔进一步缩短为 $44100/1024=43.07\text{Hz}$ 。读者也许会问,如此一来是不是就应该尽量选择更大的频率帧呢?进一步观察发现,随着频率帧的增大(频率分辨率的增加),需要更多的时域采样点。即如果选择大小为 1,024 的频率帧,在频域上将获得窄至 43.07Hz 的频率分辨率,随后需要 1,024 个时域采样点(约等于 $1024/44100=23.21$ 毫秒内的采样点数量),则比使用 32 个时域采样(采样周期为 $32/44100=0.73$ 毫秒)需要更长时域样本。换句话说,从时域的角度看,对于大小为 1,024 的频率帧,每 23.21 秒只能得到一个频谱帧,即降低时域分辨率的同时可增加频域分辨率,如图 4.9:

事实上,这是 DFT 的特性之一——即增加频域分辨率会降低时域分辨率,反之亦然。因

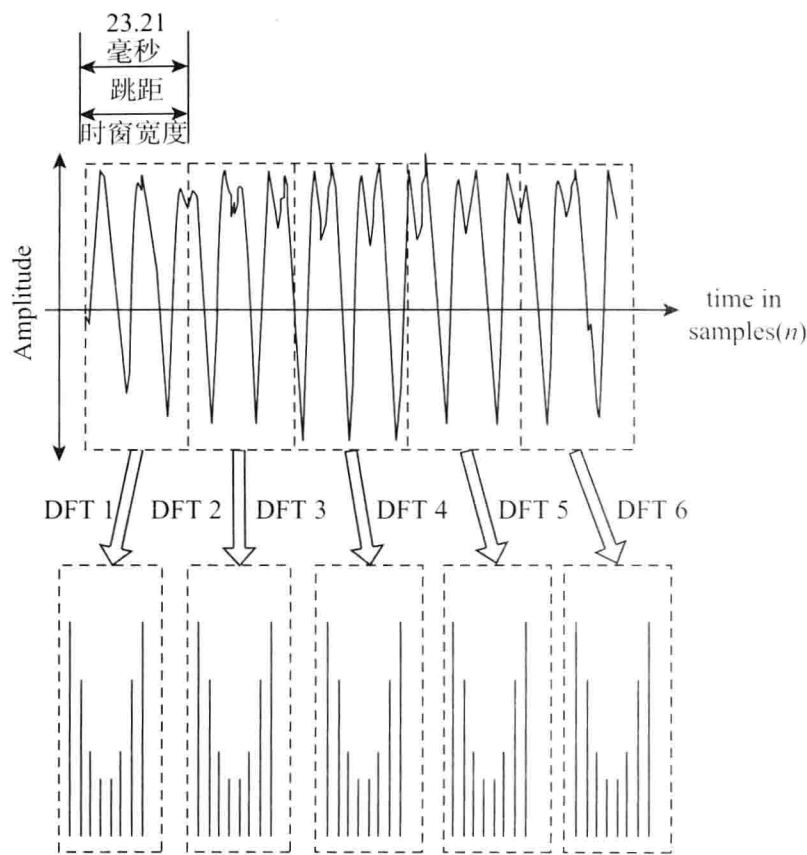


图 4.9 DFT 的时域与频域问题

此,由于两者呈反比关系,则在选择 DFT 的长度时,需要在频域分辨率与时域分辨率之间保持平衡。那么,最佳的 DFT 长度是多少呢? 在没有任何背景条件的情况下很难回答这个问题,因为答案取决于很多因素,如通过 DFT 想获得什么信息,原信号的特征,所需要的频谱细节等等。简而言之,没有确定的答案。不过,有一种算法有助于缓解频率分辨率与时域分辨率之间的矛盾,详见下节。

5. 短时傅立叶变换 STFT

DFT 本身频域和时域分辨率的矛盾有时在同时需要较高频域和时域分辨率的实际应用中会出现问题。那么,如何同时提高频域分辨率以及时域分辨率呢? 一种有效获得较好时域和频域分辨率的算法是短时傅立叶变换(short-time Fourier transform)。这个简单而又有效的算法与第 2 章第 7 节介绍的重叠算法相似。计算输入信号的两个相邻 DFT 帧时,如上节所述,窗的步进与窗本身的大小相同。即每个窗的跳距(hop size)等于窗长度,任何窗之间没有重叠。而情况并非总是如此,按照重叠算法的思想,跳距小于窗长度时,相邻两窗将重叠,相对于非重叠的情况,对于一个给定时间帧,可获得更多的频率帧。也就是说,对于给定的时长,将获得更多的频率帧,因为频率帧的大小不变,从而在不影响其频率分辨率的情况下提高了时域分辨率。重叠率越高,获得的 DFT 帧越多——则提高时域分辨率的同时没有降低频域分辨率。对于一个给定大小的跳距,时域分辨率可由式 5.1 计算,其中 hop 为跳距, f_s 为采样频率。

$$\text{time resolution} = \frac{hop}{f_s}$$

(5.1)

实际应用中,跳距往往以两窗之间的重叠比率定义。如:50%重叠率的 128 点 DFT 对应的跳距为 64 个采样点。STFT 所采用的这种重叠和时窗化的思想如图 5.1 所示,重叠率为 40%,相当于 12.73 毫秒。可通过降低跳距来进一步提高重叠率,对于同样的信号片断,其结果是增加了所计算的 DFT 频谱的帧数。STFT 算法的结果使两方面获得了较好的平衡——即较好的时域分辨率和较好的频域分辨率。由于跳距大小决定时间分辨率,而频率帧的宽度决定了频域分辨率,因此相较于非重叠时窗算法,STFT 能够同时在时域和频域获得理想的分辨率,而大多数谱分析也都普遍采用 STFT 变换,而不是单纯的 DFT 变换。

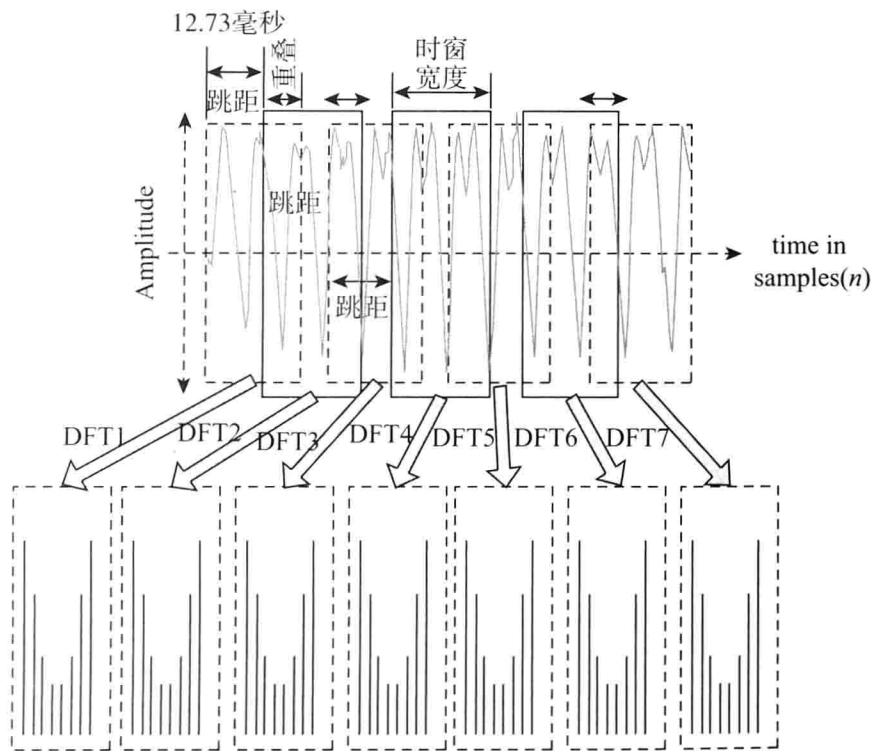


图 5.1 重叠量与跳距小于时窗宽度的 STFT

6. 零 填 充

使用一个特定大小的窗进行 DFT 变换时,较大的窗可分析较短的时域信号片断。例如,有一长度为 54 个采样点的时域信号,使用大小为 64 个采样点的窗进行 DFT 变换,频域分辨率为 $44100/64 = 689.06\text{Hz}$ 。而此例中,时域信号的采样点不足 64 个,应该如何处理呢? 可进行零填充(zero-padding)。该方法所做的是在序列的尾部用零值进行填充以补足所需的采样点,如图 6.1 所示。此例可用 10 个零点来填充原只有 54 个采样点的信号。

零填充允许使用更大的 DFT 变换长度以获得更高的频率分辨率,产生的频谱虽然对原信号频谱 $X^d[\cdot]$ 进行了插值,但由于在时域仅采用零点进行填充,因此并不意味着在新的频谱中会获得更多的信息,实际上只得到了一个有较高频域分辨率的频谱,频率轴被划分为了 64 份而不是 54 份。尤其对于快速傅立叶变换 FFT(fast Fourier transform),零填充非常有用,详见第 10

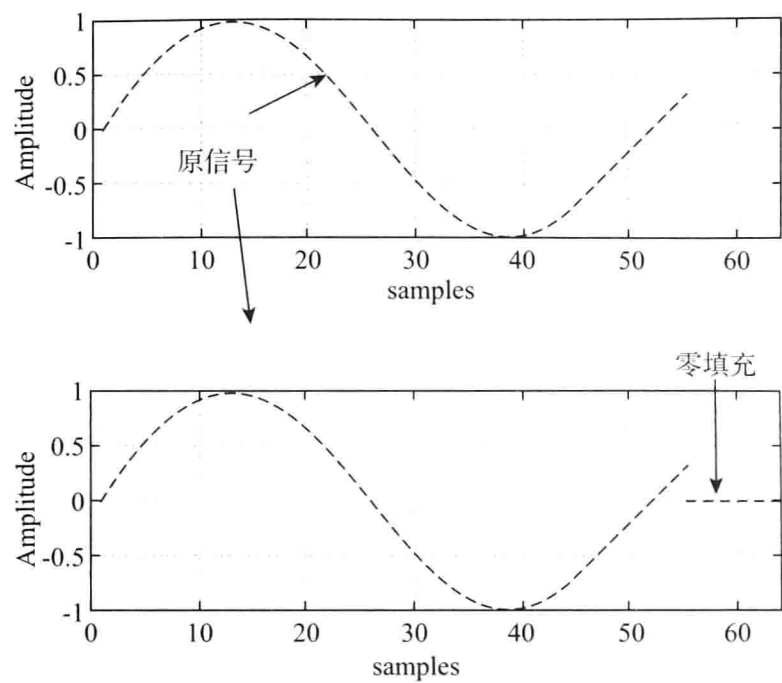


图 6.1 时域信号的零填充,非零填充(上图),零填充(下图)

节(该节还将讨论为什么时窗大小的形式为 2^N)。

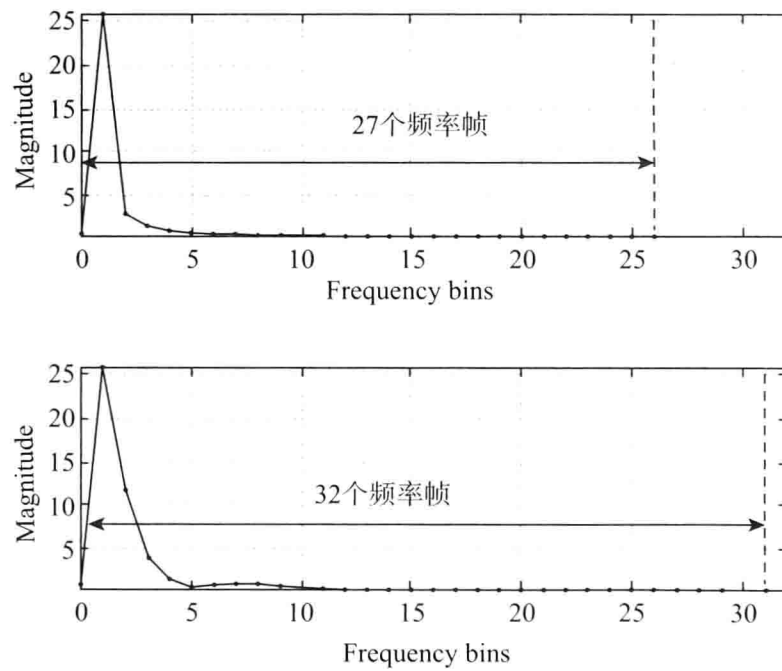


图 6.2 零填充的频谱,原信号的 DFT 频谱(上图),零填充后信号的 DFT 频谱(下图)

图 6.2 的上图显示了图 6.1 正弦波的 DFT 变换结果,DFT 分析时窗大小为 54 个采样点,下图是进行零填充,在频谱尾部增加了 5 个频率采样点后的变换结果,DFT 分析时窗大小为 64 采样点(需要注意的是,两图都只显示了原幅值频谱的一半)。由以上两图可知,除了零填充后的频谱具有更多的频率帧之外,原 54 个频率帧频谱的延展和插值,两个频谱几乎是一样的。因此,如果在时域进行零填充那么将在频域进行插值,则在频域进行零填充将时域进行插值。而事实证明确实如此。本书在此不会深入探讨这一问题,但是可以说,进行逆 DFT 变换后,插值

后的时间序列往往会产生不需要的人为失真,因此不会经常使用插值器。

7. 频率混叠的再次讨论

第1章4.2节介绍了频率混叠的概念,并在时域上阐释了此概念——即解释了发生频率混叠的后果和伪频的计算方法。第6章2.1.1节又通过频率响应分析介绍了发生频率混叠的原因,并证明了在奈奎斯特限制频率表现出的镜像幅值特征。现在,我们也有让·巴蒂斯特(Jean Baptiste)在身边,有助于更好地阐释这一重要概念。本节将从频域的角度解释频率混叠的概念。

到现在为止,所有的实例都将所关注的频率范围局限在0至 π 之间,相当于真实频率0至 $f_s/2\text{Hz}$ 。如果频率高于 $\pi, 2\pi, 3\pi, \dots$ 时,会发生什么情况呢? 回答这个问题并不困难——如前所述,可用欧拉公式来分析傅立叶变换的基本特征。即傅立叶变换FT以 2π 为周期,当 m 为整数, ω_x 是以弧度为单位的角频率时,式7.1恒成立:

$$FT(\omega_x) = FT(\omega_x + 2\pi \cdot m) \quad (7.1)$$

如果不断增加或减少角频率,得到的周期性频谱如图7.1所示:

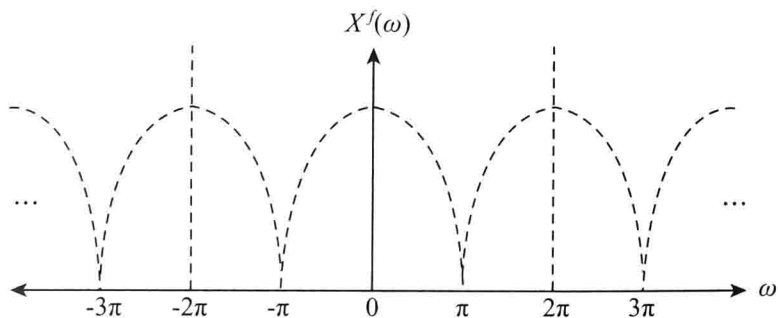


图 7.1 频谱的周期与对称特征

由图7.1可知,原始谱位于 $-\pi$ 至 $+\pi$ (或0至 2π)之间,由于频谱本身的镜像对称关系,因此在此区域内,我们只关注其内容的一半信息。此外,如果考虑 $-\pi$ 至 $+\pi$ 的原始频谱以外的区域,则会得到无限多个沿频率轴正向或负向延伸的原始谱副本。

第1章4.2节曾提到,对正弦波进行扫频,频率超过奈奎斯特限制频率时,采样后会映射至一个较低的频率。这一现象现在可用傅立叶变换进行证明。接下来使用与第1章4.2节相同的参数,设采样频率 $f_s = 100\text{Hz}$,被采样频率由1Hz、5Hz增大至20Hz、50Hz、95Hz和105Hz,则采样后信号对应的128点傅立叶变换分别如图7.2~图7.4:

正如所预期的那样,当被采样频率超过奈奎斯特限制频率时,产生频率镜像“反射”(bouncing-off)的现象,这与第1章的讨论结果相符。还可确定的是,采样后的5Hz、95Hz和105Hz正弦波之间的频谱特征差异不大,而50Hz正弦波的频谱则与0Hz信号没有区别。由图可知,与第1章提到的相同,频域幅值的极性没有变化。同时,如前所述,镜像边界位于 $\pi(f_s/2)$,其镜像图像位于 π 至 2π 或 $f_s/2$ 至 f_s 之间。如果增大或降低频率,则图像会像“悠悠球”(yo-yo ball)那样呈现出周期性的运动,如图7.1所示。

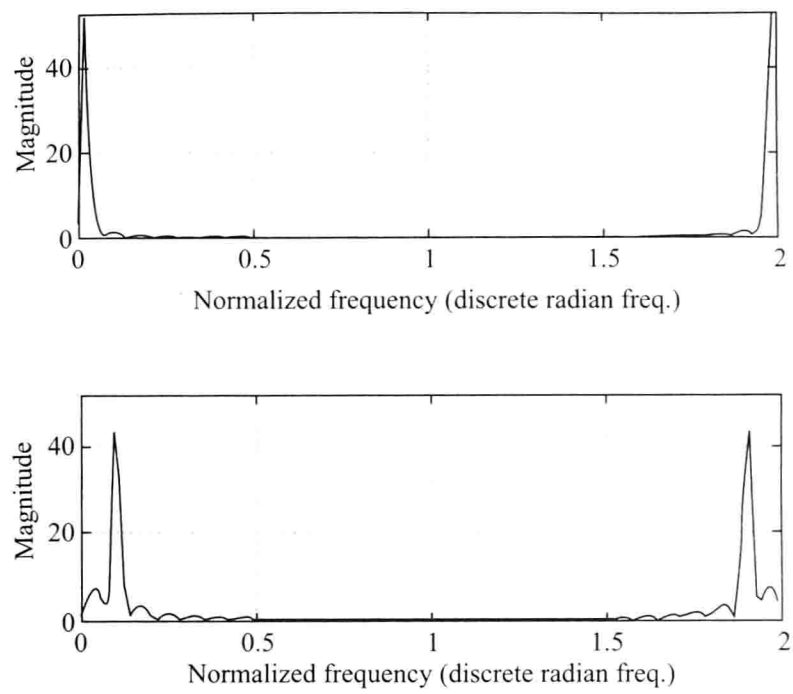


图 7.2 1Hz 与 5Hz 正弦波的频谱

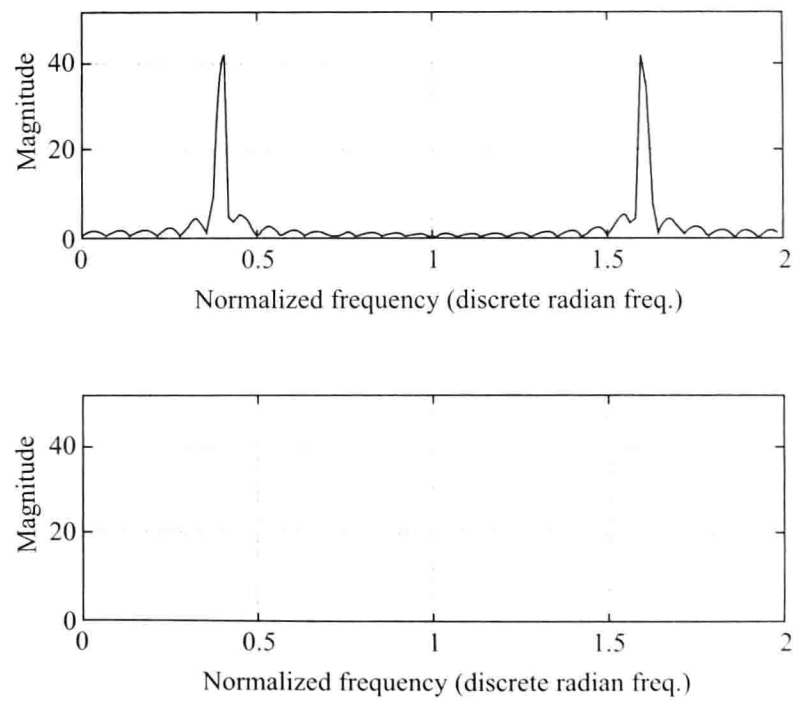


图 7.3 20Hz 与 50Hz 正弦波的频谱

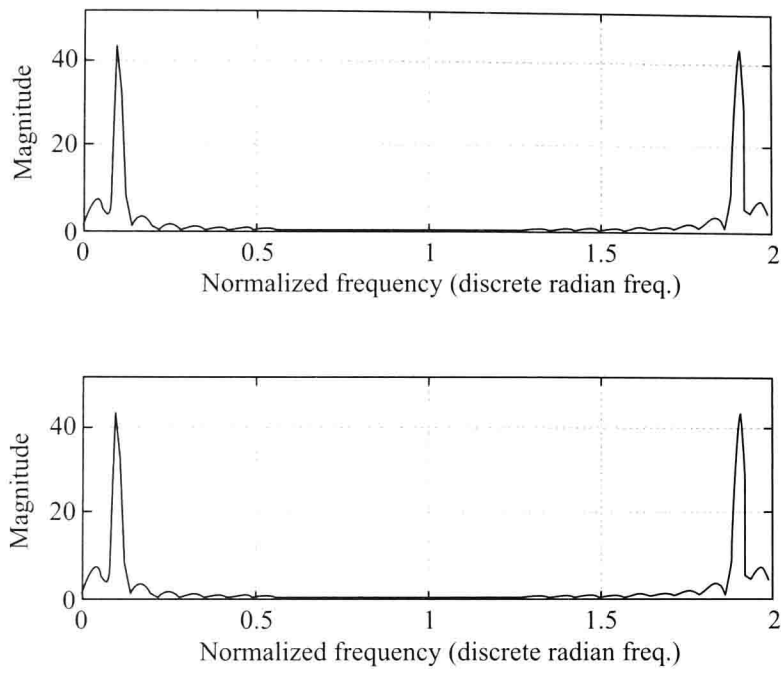


图 7.4 95Hz 与 105Hz 正弦波的频谱

8. 另一个视角的减采样与过采样

本节将在频域再次讨论减采样与过采样过程中所发生的事情,以更好地理解用低通滤波器来避免人为失真,否则会出现问题的原因。

8.1 减采样

第 1 章 6.2 节简要介绍了时域上的减采样,指减少时域信号每秒的采样点数。为了唤醒我们的记忆——设采样率为 8 kHz,对一个长度为 2 秒共有 16,000 个采样点的时间序列 $x[n]$ 进行再次采样时,总采样点数降低为 8,000,本质上对信号进行了一次因子为 2 的减采样。

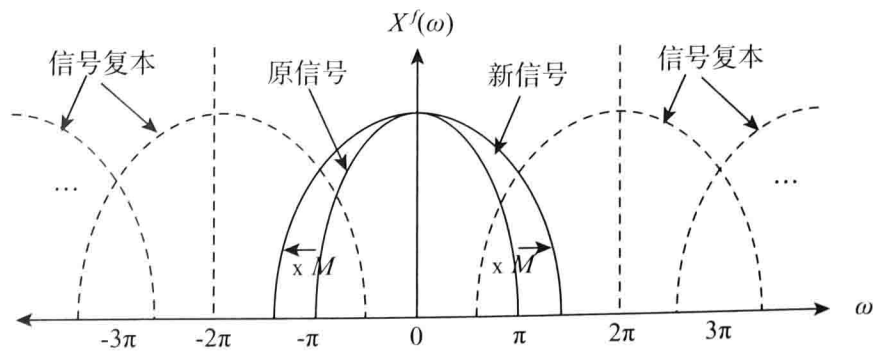


图 8.1 由减采样产生的频率混叠

接下来考虑信号的频谱,如前一节的频谱(图 7.1),以因子 M 进行减采样(M 为整数)并观察其结果。由图 8.1 可知,减采样导致原信号的频谱宽度在频率轴上由因子 M 扩展了 M 倍。

事实上,超出 $\pm\pi$ 这一奈奎斯特限制频率的部分使频谱图像在所关注的 0 至 π 区域发生了溢出和重叠。这样,超出奈奎斯特频率(π 或 $f_s/2$)的分量就发生了频率混叠,表现为一个较低的频率值! 对于由成百上千个的正弦分量组成的复合信号,降采样之后没有经过适当的低通滤波就会造成不需要的失真。为了直观地理解频谱扩展,考虑一个正弦波的周期,即频率越低,每秒的周期数越少——如 1Hz 正弦波每秒只有一个周期,采样率为 8 kHz 时,一个周期有 8,000 个采样点。对于 2Hz 的信号,相当于 1 秒钟有两个周期,则一个周期将只需要 4,000 个采样点。这样一来,减采样将使每个周期的采样点减少,从而提高了信号的频率或音调。因此,减采样过程将自动使频率增加。这一现象在第 2 章也曾提到过——即减少采样点数将提高音调(对有调信号而言)。如果对正弦波进行两倍减采样,则音高将增加一个八度。

原始信号的频率混叠可表示如下(Porat 1997):

$$X^f_{(\downarrow M)}(\theta) = \frac{1}{M} \sum_{m=0}^{M-1} X^f\left(\frac{\theta - 2\pi \cdot m}{M}\right)$$

(8.1)

其中,整数 M 是减采样因子, m 为整数,指原始谱的副本数, θ 是每采样的弧度角频率。乍一看,式 8.1 比较复杂,不过仔细观察,与所看到的相比不那么令人费解的是——等式右边满足可加性原理,减采样后的频谱是原傅立叶变换 $X^f(\theta)$ 经因子 M 进行缩放后的扩展频谱(约数为 M)。由原始谱(在 $-\pi$ 至 $+\pi$ 之间)的副本组成的净减采样频率混叠频谱所导致的结果是:频谱溢出到了相邻的由奈奎斯特限制频率的整数倍频所确定的区域。如预期的一样,求和项外的总体乘积因子 $1/M$ 保持频谱的总能量不变。 $M=2$ 时,图 8.1 的净频率混叠频谱与图 8.2 相似。表 8.1 给出了位于频点 $\pm\pi$ 处的原始谱幅值与 $m=0$ 至 1 时的减采样幅值的对比结果。

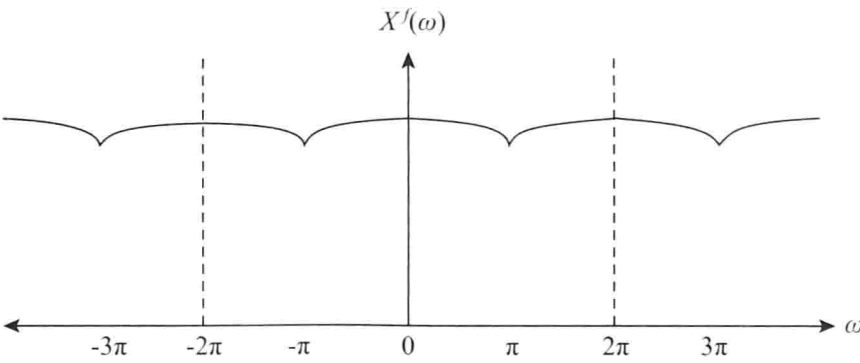


图 8.2 频率混叠后的频谱, $M=2$

表 8.1 在原始角频率 $\theta=\pm\pi$ 位置上的减采样

	频 率 (θ)	傅立叶变换 ($X^f(\theta)$)	幅 值
原始谱	$\pm\pi$	$X^f(\theta)$	0
$m=0$	$\pm 2\pi$	$X^f(\theta/2)$	0
$m=1$	$4\pi, 0$	$X^f((\theta-2\pi)/2)$	0

为了避免在减采样时发生频率混叠,需要在减采样或数字化前对输入信号低通滤波。如果在采样过程之后再进行低通滤波,则已经发生了频率混叠(如果信号存在高于奈奎斯特频率的

分量)。由于因子为 M 的减采样使频谱带宽增加 M 倍,则需要的低通滤波器的截止频率为 π/M 或更低。之所以考虑使用更低的截止频率完全是因为真实的滤波器不是第 7 章所述的砖墙式滤波器。因为每个真实滤波器都具有某种类型过渡带,所以设计滤波器时需要考虑安全余量。为了避免频率混叠,减采样须按如下过程进行,如图 8.3。有时该过程也称为抽样(decimation)而不是减采样,后者指不经低通滤波直接删除样点,而抽样则指经低通滤波后再进行减采样。

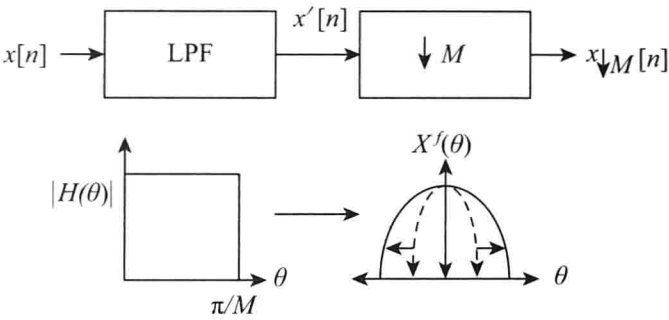


图 8.3 减采样前的低通滤波器

在某种意义上说,如果从时域连续模拟信号数字化的角度来看,可更直观地理解减采样后,原信号中超过奈奎斯特频率的分量发生频率混叠率的原因。由时间连续模拟信号的采样过程可知,模拟信号中高于奈奎斯特频率的分量会发生频率混叠。也就是说,时域连续采样本质上是对模拟信号进行“减采样”——模拟信号的时域及频域分辨率是无限的,而数字信号则是有限的(或大或小的带宽)。因此,采样后的减采样是对采样后信号(时间离散信号)的另一种形式的重采样。在被减采样过程中,如果最高频率分量小于 π/M ,所有分量都不会发生频率混叠,不需要在采样前加入低通滤波器。

8.2 过采样

对信号进行过采样时,会增加采样点的数目,并对原信号进行交织零填充。例如,进行因子为 2 的过采样,就是在两个原相邻采样点之间插入一个零点。在过采样[也称为时域上的扩展(expansion)]过程中,原频谱图 7.1 变为了新的频谱图 8.4。

正如我们所估计的那样,过采样与减采样相比,频谱不是扩大了而是缩小了。由图 8.4 可知,过采样同样会产生不希望得到的萎缩频率。因为频谱发生了萎缩,产生了不需要的人为失真,所以在过采样后使用一个截止频率为 π/L 的低通滤波器可消除由附加零点带来的频谱失真。对于此例,过采样的过程如图 8.5 所示。

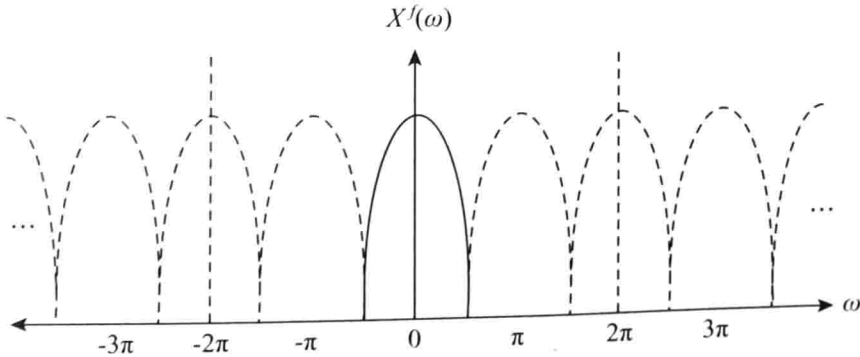


图 8.4 过采样,因子 $L=2$

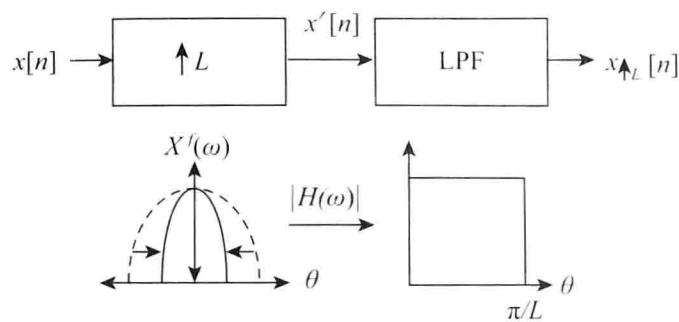


图 8.5 过采样与频点为 π/L 的低通滤波

不难理解,低通滤波器能够解决过采样时插入零点带来的问题——使信号更平滑。一个零交织填充信号就如“坑坑洼洼”(potholed)的数列一样,在频域上消除了失真,在时域上就会变得更加平滑。图 8.6 给出了一个采样率 $f_s=40\text{Hz}$, 频率为 1Hz 的离散信号的过采样过程,上图为原始信号,中图为 $L=2$ 的零交织插值后的过采样信号,下图为经低通滤波后的过采样信号,此例采用的是二阶巴特沃斯低通滤波器。对于过采样声信号的特殊音质,听到的是带宽的降低由傅立叶变换结果可知——频谱向 DC 直流分量萎缩,而高频分量的减少也让声音听起来不那么明亮并略显沉闷。

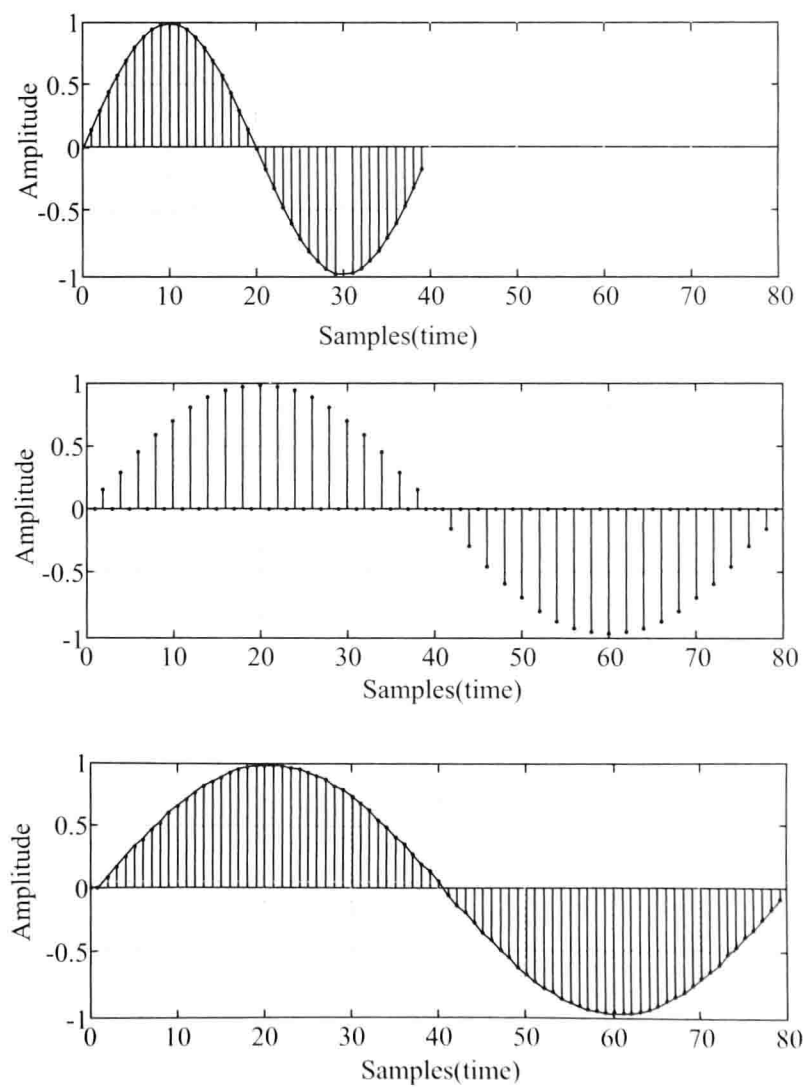


图 8.6 原信号(上图),零交织插值后的过采样信号(中图, $L=2$),以及低通滤波后的过采样信号(下图)

9. 时窗的频域分析

无论出于什么目的进行信号处理,几乎都会用到时窗分析。迄今为止,已经在 RMS 计算、OLA 算法、傅立叶变换、STFT 变换以及其他场合使用了时窗分析的概念。如第 2 章第 4 节所述,不同的窗口类型有不同的时域特征和时域图像。矩形窗是最简单的、具有尖锐边缘的时窗,而如汉恩窗则是两侧逐渐衰减的时窗,使信号在时域上产生渐入和渐出效果。

不同的时窗形状在频域上也有不同的波动纹理特征。因此,为了减少波动 DFT 变换时窗参数带来的失真——进行 DFT 时需要选择合适的时窗类型。更具体地说,在频域上决定时窗性质和行为的主要参数是主瓣(main lobe)、旁瓣(side lobe)和滚降特性(roll-off)。接下来以 32 点矩形窗为例,了解这 3 个重要参数对于时窗选择的意义。

这 3 个典型的特征参数决定了时窗的类型,如图 9.1 的矩形窗,主瓣基本以离散峰的最小间距(频率上接近的峰值相当于两个正弦分量)决定了信号的分辨率。主瓣宽度越小,越有利于得到更高的频域分辨率,同时也有助于区分并分离其他接近的频率分量。主瓣以外的其他频段的峰值称为旁瓣。旁瓣特征也是时窗类型的函数,信号的声染色和失真程度主要由旁瓣的幅值强度决定。相对于主瓣,旁瓣能量会溢出至相邻的频率帧。如果旁瓣能量过高,可能会导致邻近频段的较小峰值被掩蔽。滚降特征则表现了旁瓣幅值能量的衰减特性。因此,理想的时窗将具有一个较窄的主瓣和幅值较低的旁瓣。

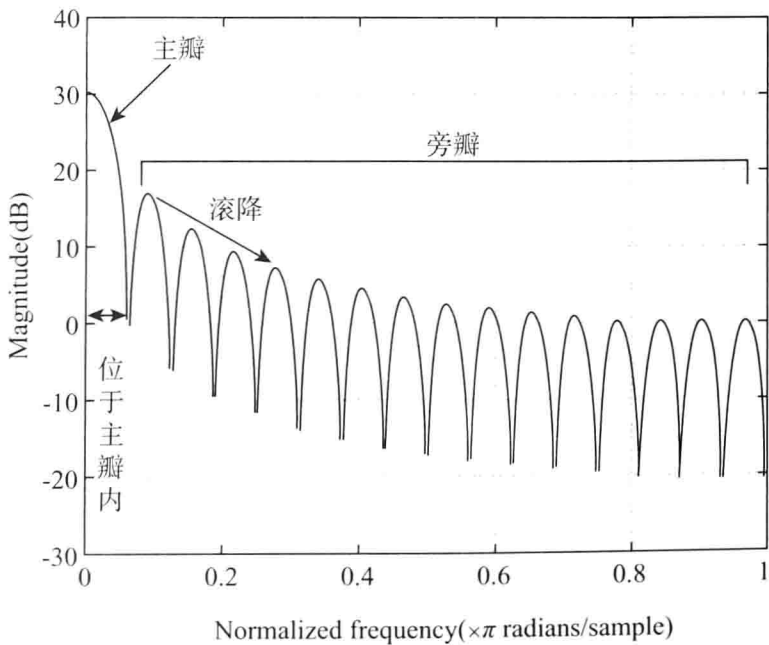


图 9.1 矩形窗的频域特征

为了更好地理解上述观点,下面以两列振幅为 1.0 的正弦波和零填充为 1,024 点的 32 点矩形窗为例进行 DFT 变换。零填充后的矩形窗如图 9.2,对每个正弦波进行时窗分析,每个时窗只截取正弦波的前 32 个采样点。为了更好地观察 DFT 变换后主瓣和旁瓣对两个正弦波频域特征的影响,此例采用了足够小的时窗长度(幅值为单位增益)。

接下来的图 9.3 和图 9.4 分别给出了由上述矩形窗截取后的 500Hz 正弦波和 1 kHz 正弦波的频域表现。由以下两图可清楚地确定和识别主瓣与旁瓣——具有独特宽度的主瓣位于并包的频域表现。

裹在中央峰值附近,表示了正弦波的频率,旁瓣逐渐向两侧衰减。现在,将两个频率分别为 500Hz 与 1 kHz 的正弦波进行叠加,会发生什么情况? 叠加后的复合音如图 9.5 所示。尽管仍然可区分两个分别对应 500Hz 和 1 kHz 正弦波的峰值,但因两列时窗化正弦波的旁瓣能量溢出的净累积而产生了失真。

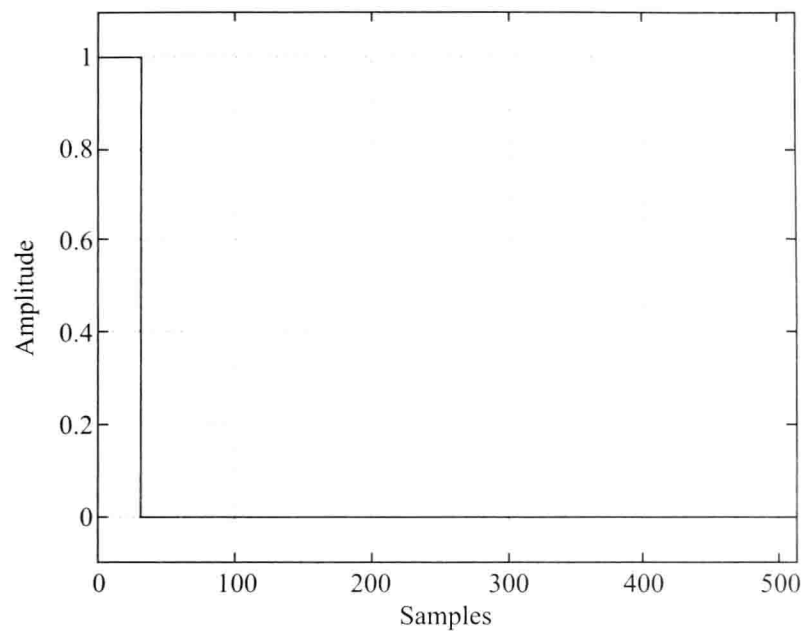


图 9.2 32 点矩形窗以零填充为 1,024 点

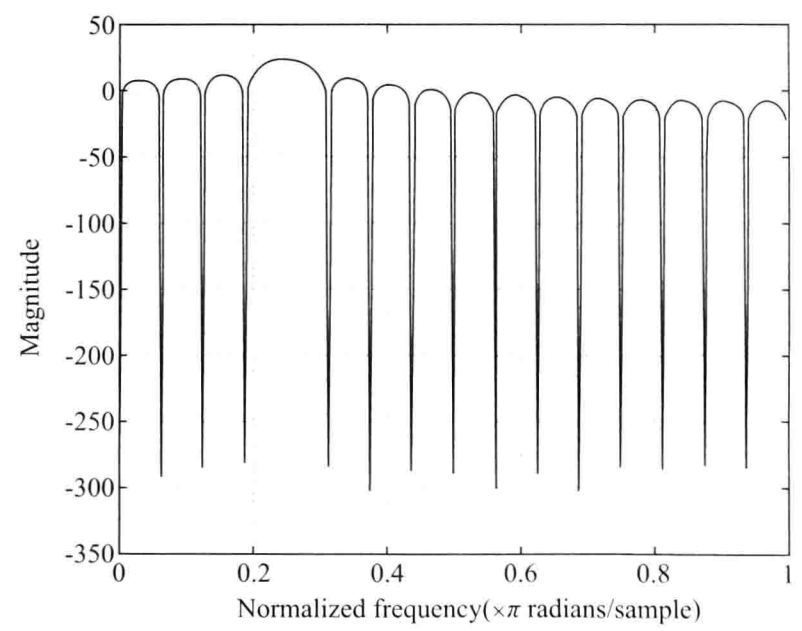


图 9.3 由矩形窗截取的 500Hz 纯音的 DFT 变换

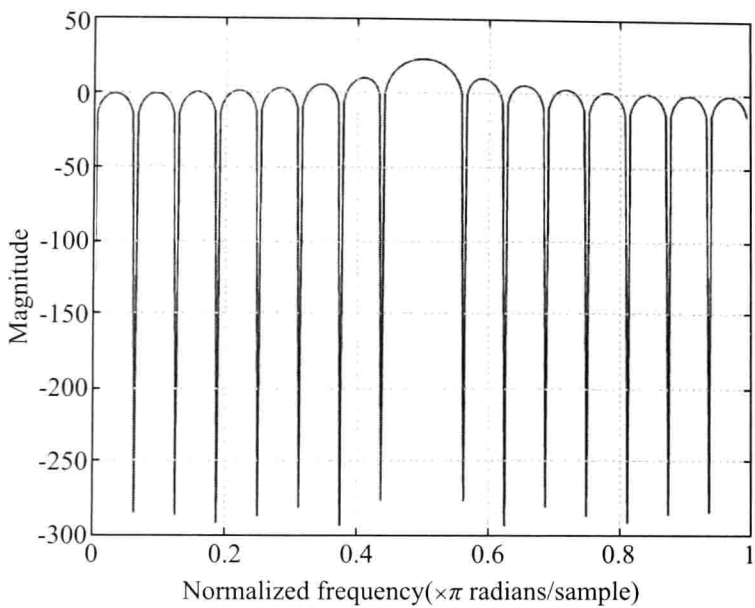


图 9.4 由矩形窗截取的 1,000Hz 纯音的 DFT 变换

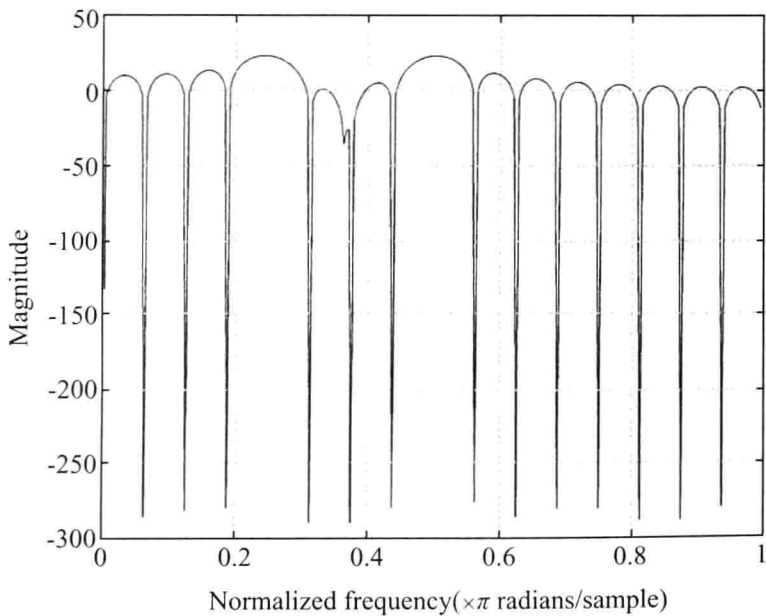


图 9.5 由矩形窗截取的 500Hz 和 1,000Hz 复合音的 DFT 变换

现在以幅值较小的 500Hz 正弦波为例,如图 9.6,使 500Hz 正弦波的幅值降低至原来的 10%,保持 1 kHz 正弦波的幅值不变。由图可知,DFT 变换的结果更加模糊,也更难判断到底发生了什么情况——500Hz 正弦波的峰值不再突出,几乎被掩蔽在了 1kHz 正弦波的旁瓣中。音乐信号和声音信号往往远比此例的波形复杂,有很多突出的峰值和谐波,是一个复合信号。这就使峰的识别更加困难,因为其中有时窗选择不当所带来的失真。因此,一般情况下,选取时窗时,要使旁瓣幅值保持在尽可能低的水平,这不仅是为了缓解频谱能量泄漏的问题,也是为了使主瓣本身的峰值更为明显和突出。

下面用最后一个例子说明主瓣和旁瓣的重要性,将合成复合音的两个正弦波频率调得更近,如将 500Hz 提高到 920Hz,幅值仍为 1.0,结果如图 9.7。尽管希望从图中看到分别对应两个正弦波频率的幅值,但结果是只有一个大而宽的峰值——有点类似影片《第三类接触》

(*Encounters of the Third Kind*) 中山峰形状。无论如何,因为主瓣的特性,920Hz 分量(或 1 kHz 分量)在处理过程中已经损失。因此,最重要的一点是,在观察频谱和分析数据时,至少要了解频谱中主瓣和旁瓣特征的相互影响,以对所得频谱的正确性作出评估。

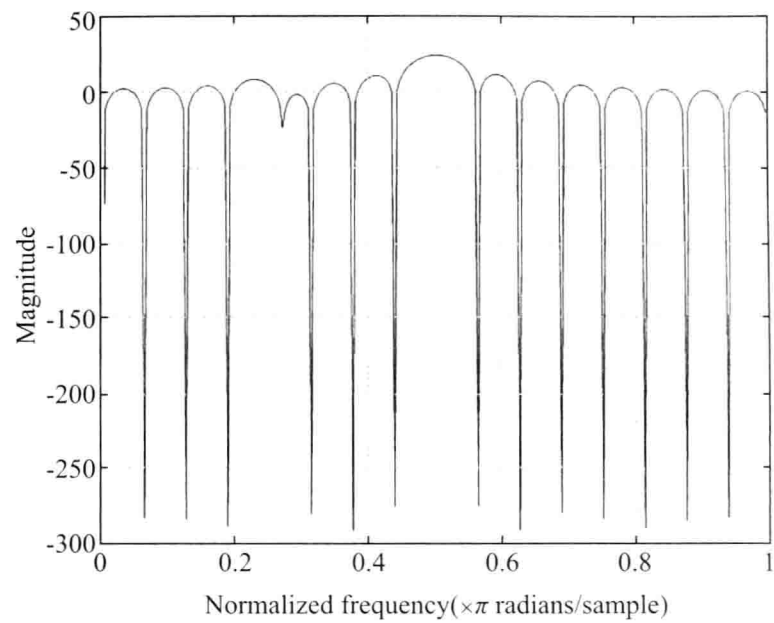


图 9.6 由矩形窗截取的幅值分别为 0.1 和 1 的 500Hz 和 1,000Hz 纯音复合音的 DFT 变换

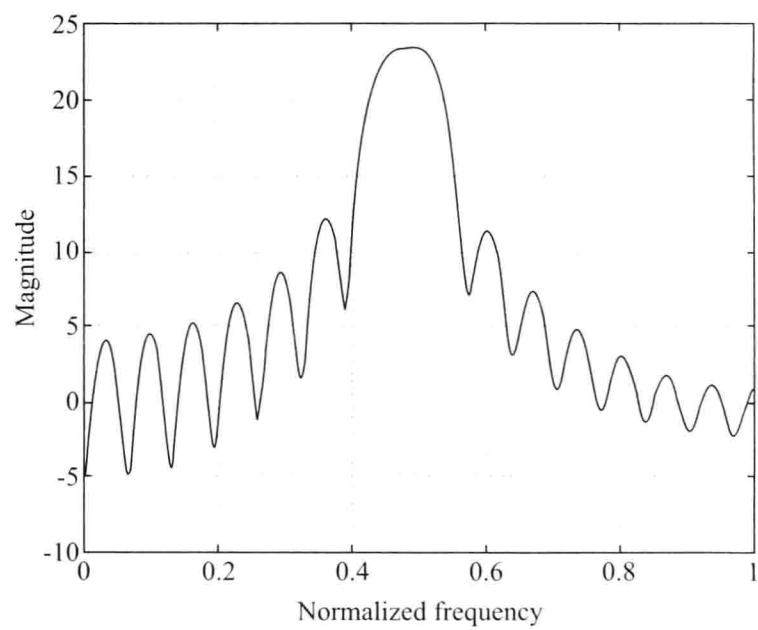


图 9.7 由矩形窗截取的 920Hz 和 1,000Hz 纯音复合音的 DFT 变换

如本章第 4 节所述,提高频域分辨率的简单方法是使用大长度的时窗(DFT 变换长度)。然而,这一方法并不能解决所有由旁瓣带来的问题。因此需要根据信号分析或处理的目的,选择合适时窗的类型。一些具有代表性的时窗类型及其主要特点将在下一节讨论。

9.1 矩形窗

矩形窗是最常见的一种时窗类型,时窗本身即是一个集合——整个时窗内的幅值具有单位增益,如图 9.8。其频域主瓣宽度为 $4\pi/N$ (N 为采样点数),第一旁瓣衰减量为 -13.3dB ,滚降率

为 20 dB/倍频(dB/decade)。由于旁瓣能量较高,矩形窗也存在较小的主瓣能量溢出因子。与此同时,因其主瓣狭窄,相对来说非常适合分析瞬态信号——即幅值变化较快的信号。

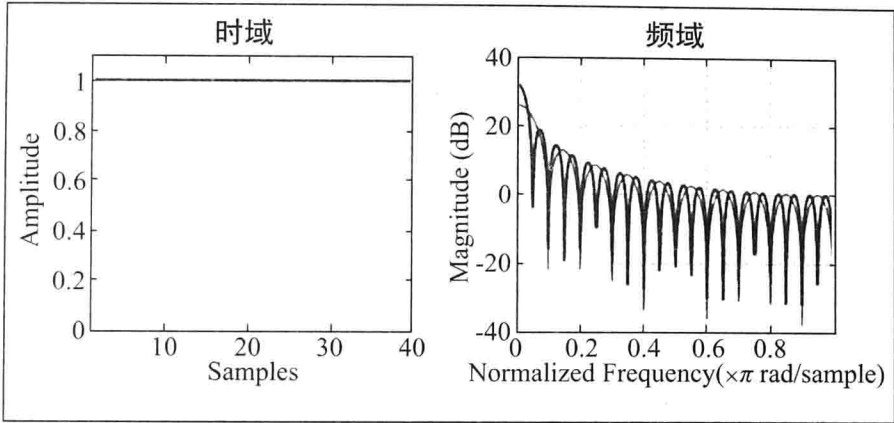


图 9.8 矩形窗

9.2 汉恩窗

汉恩窗有时也称为汉宁窗,旁瓣幅值以加和方式衰减。该时窗边缘的衰减特征可通过 3 个狄利克雷内核(Dirichlet kernel)进行平移叠加实现,狄利克雷内核的定义见 9.6 节。由汉恩窗的特点可知,它有时也被称为余弦窗,它的第一旁瓣衰减量为-32 dB,主瓣宽度为 $8\pi/N$,滚降率为 60 dB/倍频。正是由于较大的滚降率和较宽的主瓣,其主瓣能量溢出因子小于矩形窗口。

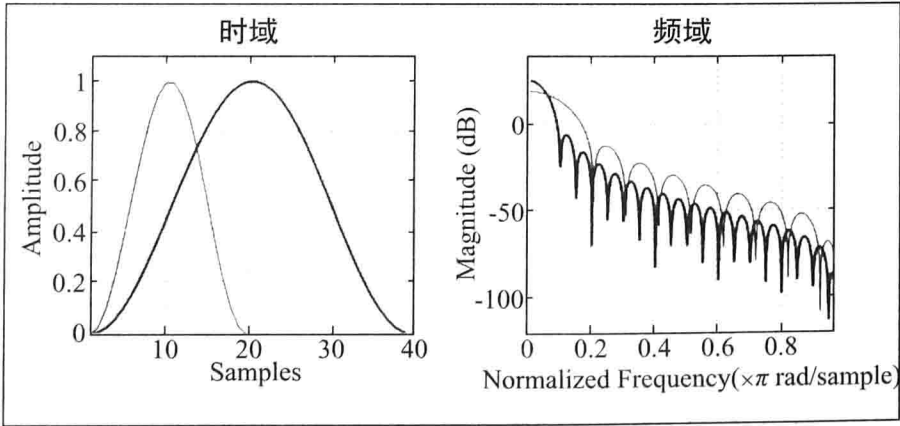


图 9.9 汉恩窗

9.3 汉明窗

汉明窗与汉恩窗类似,由多个加权狄利克雷内核修改而成。汉明窗的时域和频域特点如图 9.10所示,频域主瓣宽度为 $8\pi/N$,它的第一旁瓣衰减量为-43 dB,滚降率为 20 dB/倍频。汉明窗的一个显著特点是,窗的头部和尾部是非零值,有时也被称为半升余弦窗。由于汉明窗有尖锐的边缘,像一个矩形窗与汉恩窗的叠加,有较窄的主瓣和较好的瞬态特性。有趣的是,汉明窗的第一旁瓣比第二旁瓣要小。汉明窗同时也是声音信号频谱分析和声音合成中广泛应用的时窗类型。

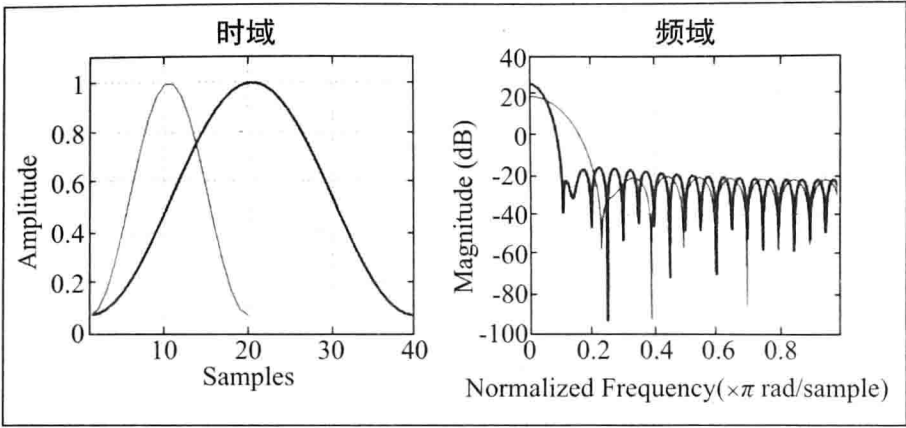


图 9.10 汉明窗

9.4 布莱克曼窗

布莱克曼窗 (Blackman window) 的第一旁瓣衰减量为 -58 dB , 主瓣宽度较大, 为 $12\pi/N$, 滚降陡峭, 滚降率为 60 dB/倍频 , N 是单位为采样点的时窗长度, 如图 9.11 所示。虽然较之与其他时窗的图像, 布莱克曼窗的主瓣较宽, 但旁瓣落在频谱的低频段。

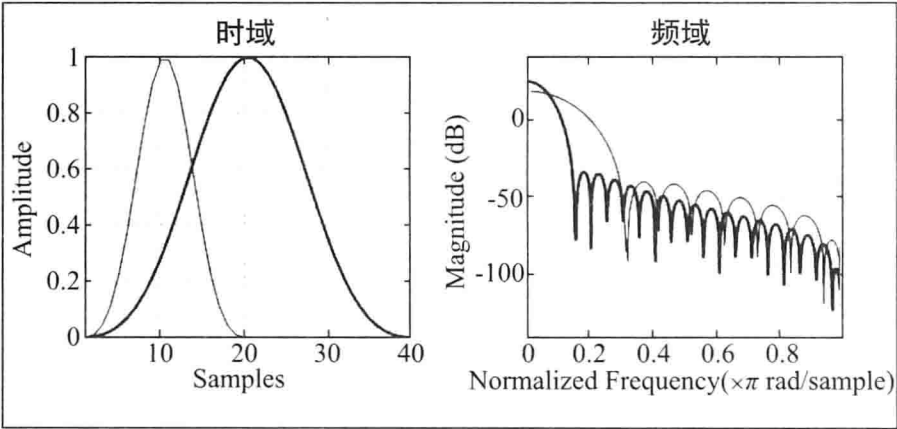


图 9.11 布莱克曼窗

9.5 切比雪夫窗与凯瑟窗

有些时窗, 如切比雪夫窗 (Chebychev window) 和凯瑟窗 (Kaiser window) 拥有更直接的参数可用来控制窗的形状。如, 旁瓣衰减量 R 用于在切比雪夫窗中控制旁瓣相对于主瓣的大小, 不同的 R 值所对应的时域与频域窗口形状如图 9.12 和图 9.13 所示:

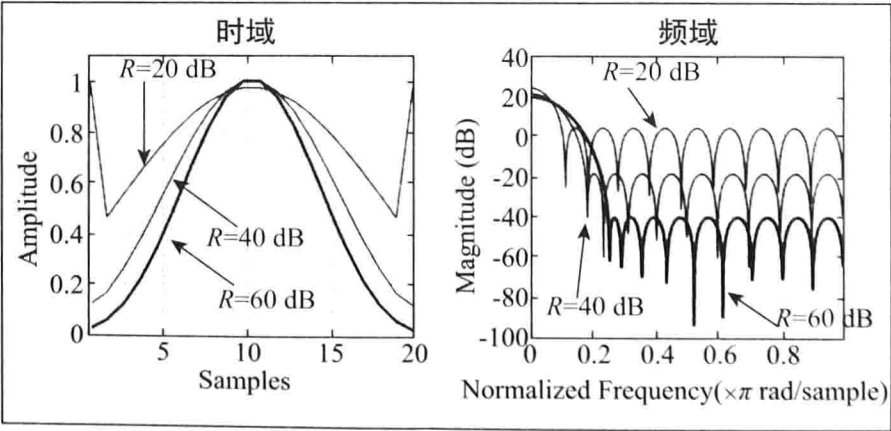


图 9.12 切比雪夫窗, $N=20$

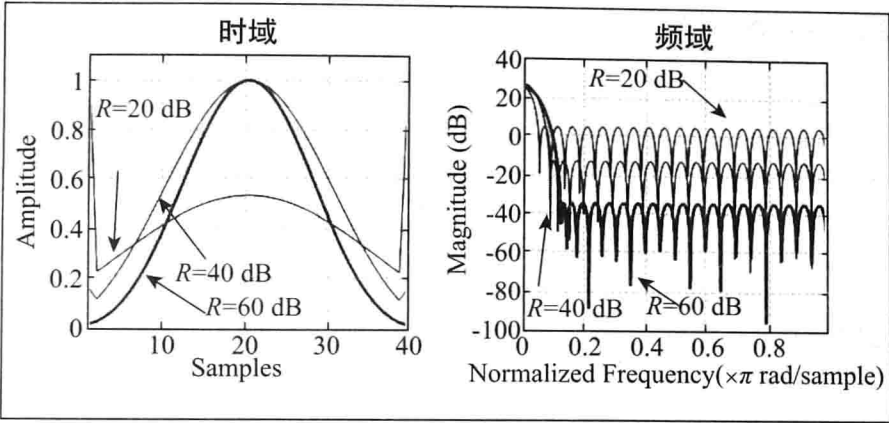


图 9.13 切比雪夫窗, $N=40$

对于凯瑟窗,通常用系数 β 来决定旁瓣的衰减,如图 9.14 和图 9.15,分别是长度 $N=20$ 和 $N=40$ 的凯瑟窗在不同系数下的时域与频域窗口形状。

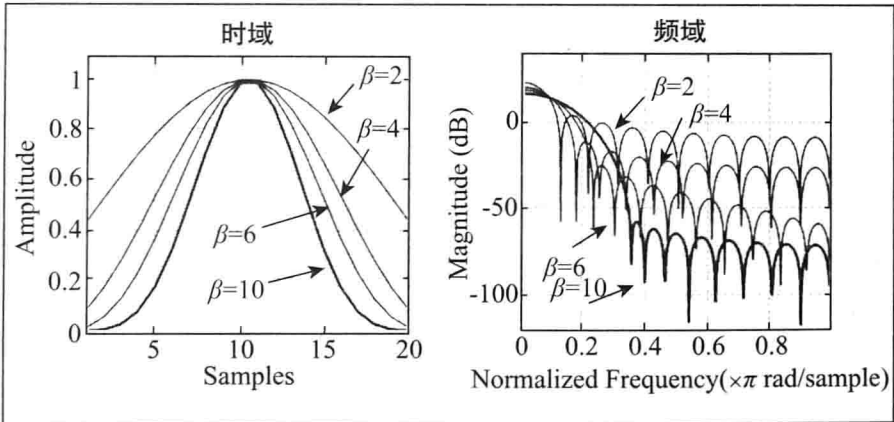


图 9.14 凯瑟窗, $N=20$

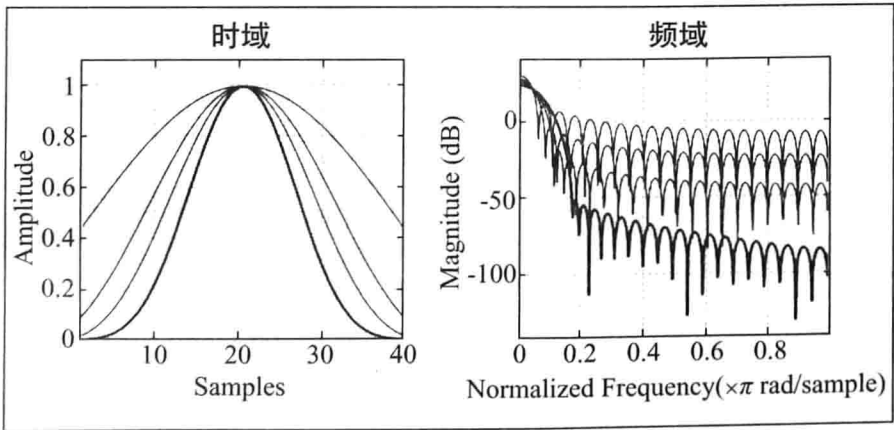


图 9.15 凯瑟窗, $N=40$

正如本节所述,各时窗类型有不同的特点,因此,选择时窗时重要的是选择一个满足既定使用要求和信号类型的时窗。

9.6 时窗的其他问题

希望到此,我们能够对时窗类型和各种时窗的特点有一个很好地理解,特别是时窗类型与频谱失真之间的关系有助于选择正确的时窗类型。在结束有关时窗问题的讨论之前,最后以矩形窗为例,详细讨论决定时窗频域特性的 sinc 函数。设有一矩形窗 $w[n]$ 的定义如下:

$$w[n] = \begin{cases} 1, & 0 \leq n \leq N-1 \\ 0, & \text{其他} \end{cases} \quad (9.1)$$

$x[n]$ 和时间离散傅立叶变换 DTFT 如式 9.2, 其中 θ 是数字角频率, 单位为弧度/采样。

$$X^f(\theta) = \sum_{n=0}^{N-1} w[n] \cdot x[n] \cdot e^{-j\theta n} \quad (9.2)$$

因为矩形窗在采样点 0 至 $N-1$ 之间具有单位增益, 则对于所有 n 的值, 令 $x[n] = 1$, 利用几何级数求和公式, 式 9.2 可写成:

$$\begin{aligned} X^f(\theta) &= \sum_{n=0}^{N-1} 1 \cdot 1 \cdot e^{-j\theta n} \\ &= \sum_{n=0}^{N-1} (e^{-j\theta})^n \\ &= \frac{1 - e^{-j\theta N}}{1 - e^{-j\theta}} \end{aligned} \quad (9.3)$$

用欧拉公式将分子和分母改写为正弦和余弦分量, 有:

$$\begin{aligned} X^f(\theta) &= \frac{1 - e^{-j\theta N}}{1 - e^{-j\theta}} \\ &= \frac{(e^{j\theta N/2} - e^{-j\theta N/2}) \cdot e^{-j\theta N/2}}{(e^{j\theta/2} - e^{-j\theta/2}) \cdot e^{-j\theta/2}} \\ &= \frac{(e^{j\theta N/2} - e^{-j\theta N/2})}{(e^{j\theta/2} - e^{-j\theta/2})} \cdot e^{-j\theta(N-1)/2} \\ &= \frac{\left(\frac{e^{j\theta N/2} - e^{-j\theta N/2}}{2j} \right)}{\left(\frac{e^{j\theta/2} - e^{-j\theta/2}}{2j} \right)} \cdot e^{-j\theta(N-1)/2} \\ &= \frac{\sin(\theta \cdot N/2)}{\sin(\theta/2)} \cdot e^{-j\theta(N-1)/2} \end{aligned} \quad (9.4)$$

则函数:

$$D(\theta, N) = \frac{\sin(\theta \cdot N/2)}{\sin(\theta/2)} \quad (9.5)$$

定义为狄利克雷内核。对于 $\theta = 0 \sim 2\pi$, $N = 20$ 和 $N = 40$, 幅值曲线分别如图 9.16 和图 9.17。

我们所关注的频率定义域为 0 至 π , 并去除频率高于 π 或 $f_s/2$ 的幅值。同时也注意到, 式 9.5 与 sinc 函数非常相似, 像一个归一化的 sinc 函数, 定义如式 9.6, 频域图像如图 9.18。

$$\text{sinc}(x) = \frac{\sin(\pi \cdot x)}{\pi \cdot x}$$

(9.6)

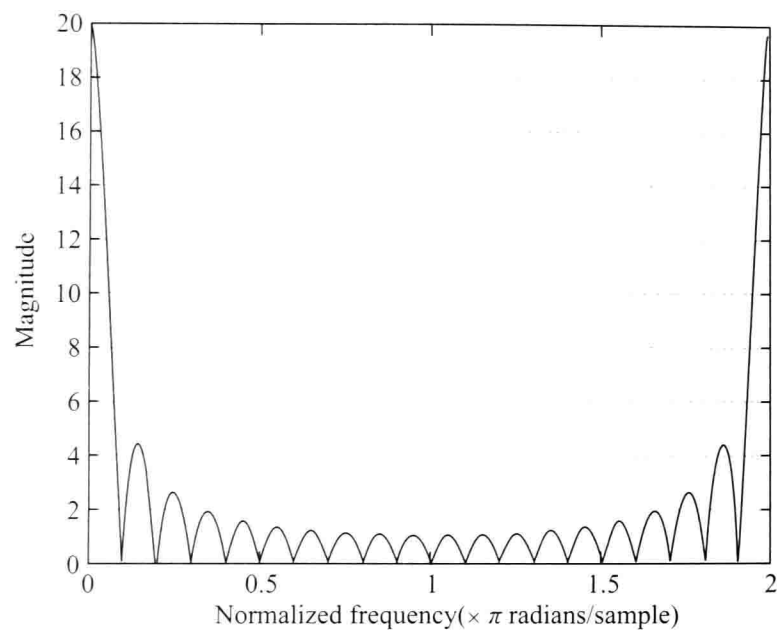


图 9.16 Dirichlet 函数 0~2 π 的频谱, $N=20$

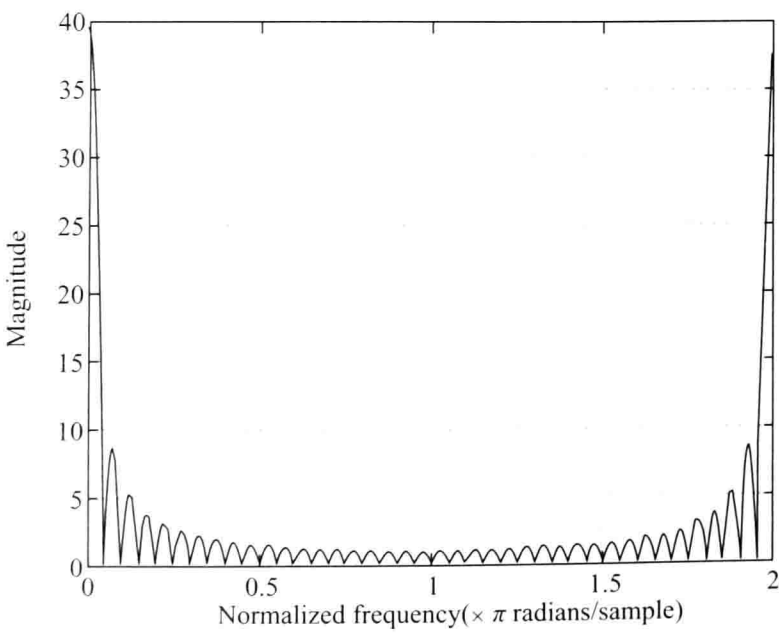


图 9.17 Dirichlet 函数 0~2 π 的频谱, $N=40$

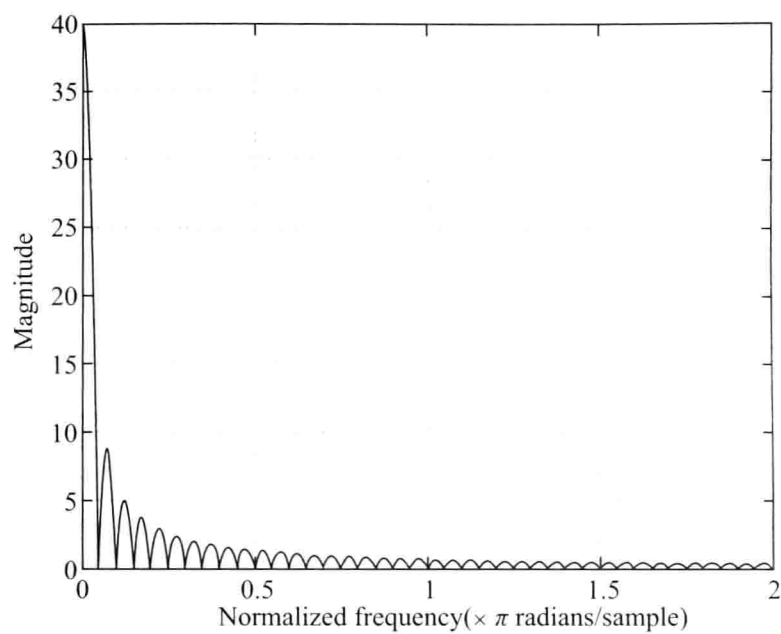


图 9.18 Sinc(x) 函数

由图 9.17 可知,sinc(x)函数的图像与 0 到频率轴中心的狄利克雷内核图像类似。但经仔细观察可知,两者差异在于频率增大至 π 值以后的区域。频率超过 π 值之后,狄利克雷内核的频谱本身呈镜像对称关系,而 sinc(x) 函数的频谱幅值随频率增大而衰减,如图 9.19:

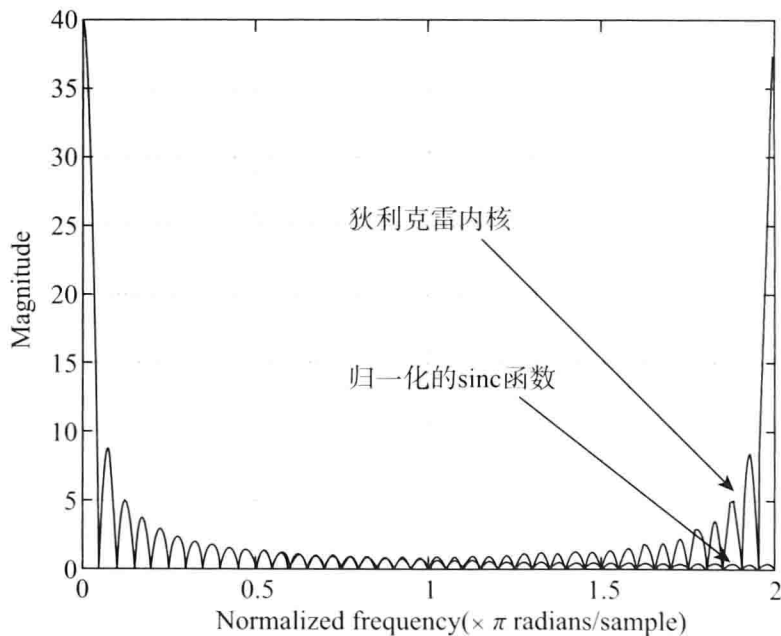


图 9.19 归一化后的 Dirchlet 函数与 Sinc(x) 函数频谱

一个很重要的特点是,在数字信号处理中,狄利克雷内核表征了频域上过零点的位置。也就是说,频率轴上的第 1 个零点对应着时域矩形窗基频所对应的一个完整波动周期。窗宽度越大,第 1 个零点的频点(频率窗的中心频率)越低。上例采用 $N=40$,频域上第 1 个零点的频点对应着与该时窗宽度等长的一个完整周期。因此,时窗宽度 N 越大,第 1 零点的频点越低(时间与频率呈反比关系 $f=1/t$)。这也是为什么时窗宽度越大主瓣越窄的原因。随后的零点位置出现在第 1 个零点的谐波上,如第 2 零点对应矩形窗的二次谐波,第 3 零点对应矩形窗的三次谐波,依次类推。

如本节一开始所述,主瓣对应着波动的基频分量,旁瓣是镶边频率。这种现象可用下式表示,其中复合波动信号为 $x[n]$ 。

$$x[n] = A \cdot e^{j\omega_x n} \quad (9.7)$$

如果以时窗 $w[n]$ (定义域为 $n=0 \cdots N-1$) 计算 0 至 $N-1$ 内的 DTFT 变换 $X^f(\theta)$, 有:

$$\begin{aligned} X^f(\theta) &= \sum_{n=0}^{N-1} w[n] \cdot x[n] \cdot e^{-j\theta n} \\ &= \sum_{n=0}^{N-1} w[n] \cdot A \cdot e^{j\theta_x n} \cdot e^{-j\theta n} \\ &= A \cdot \sum_{n=0}^{N-1} w[n] \cdot e^{-j(\theta - \theta_x) \cdot n} \\ &= A \cdot X_w^f(\theta - \theta_x) \end{aligned} \quad (9.8)$$

由式 9.8 可知,复合波信号由时窗 $w[n]$ 截取的 $x[n]$ 的 DTFT 变换是平移后的 $w[n]$ 的 DTFT 变换,频移量为 θ_x 。这就说明了为什么本章一开始提到主瓣对应的是信号的基频,且相对于时窗基频的频移量为 θ_x 。

10. 快速傅立叶变换

如今,很多音乐家在演出现场、创作工作室、实验室和家中用傅立叶变换对声音进行实时替换、修改、分析和合成。随着个人计算机成本的不断下降,CPU 处理器的速度越来越快,DFT 已经变得无处不在。而更早时需要一个房间的全部硬件才能完成的运算,如今一台多核家用电脑就可以完成,当时很难想像一二十年后的情况。由于巨大的冗余量,那时用机器计算 DFT 变换非常困难,耗时很长。直到 1965 年,库利(Cooley)和特奇(Tukey)发明了更快捷的算法时,情况才有所改观(Cooley, Tukey 1965)。这一革命性的算法被称为快速傅立叶变换(fast Fourier transform),简称 FFT。由于互联网及数字信号处理的书籍有大量关于 FFT 的介绍,本书就不再详细讨论,只简要回顾一下该算法的基本思想和概念。简要地说,库利和特奇所发现的是,DFT 变换的长度 N 为非质数时,可采用分治法(divide-and-conquer method)来进行 DFT 的快速计算。可将一个 DFT 变换分解成多个较短的子 DFT 变换,然后再将这些子 DFT 变换结果组合在一起,相对于直接用式 4.10 进行计算所需要的时间更少。更重要的是,当 DFT 的大小被限制为 2 的幂函数时,计算阶数将大幅下降,由直接计算 DFT 时所需的 N^2 阶降至 $N \cdot \log_2 N$ 阶。这一算法将傅立叶变换分割为两个更小的长度为 $N/2$ 的子变换,即每步分解后的长度都以 2 为基数(radix-2,这也可能是一个很酷的乐队名字),这也是已知使用最为广泛的 FFT 算法之一。而在之前的 DFT 例子中,窗长度总是 2 的整数次幂,这也是该算法能大大提高计算速度的原因之一吧!

计算 FFT 时,对每一子变换计算结果的组合与重排可应用到下一个子分解中——有点像循环序列,这种排列叠加的方法被称为蝶形配置(butterfly configuration)——这里,蝶形配置不是什么特别的跆拳道技术,而是指一种算法。图 10.1 给出了一个三级 8 点的 FFT 计算图,每一级看上去都像一群飞舞的蝴蝶,该算法因此而得名。值得注意的是,在该算法的流程图中,每一步的衔接和循环叠加都非常紧凑,结构相当完美。

图 10.1 采用了符号 W_N ,使图形更为简洁。由式 4.13 可知, W_N 定义为 DFT 中的指数项,如下所示:

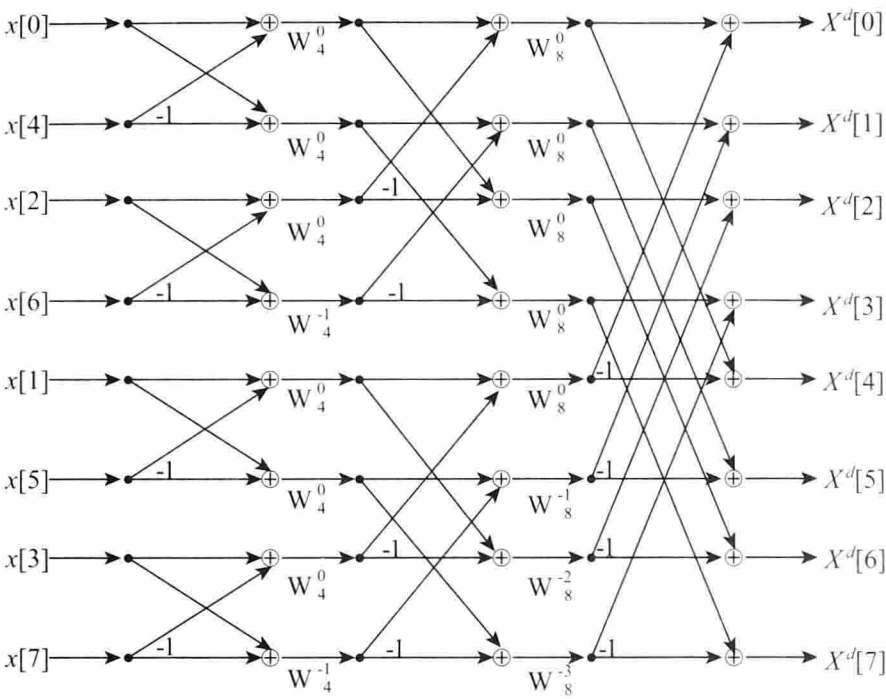


图 10.1 三级 8 点 DFT 计算蝶图

$$W_N = e^{j2\pi/N}$$

(10.1)

该指数项符合指数幂运算的乘法定律,则它的 M 次幂为式 10.2:

$$W_N^M = (e^{j2\pi/N})^M = e^{j2\pi M/N}$$

(10.2)

11. 卷积的再讨论

第 5 章第 5 节介绍了卷积及其在音频软件中的应用,其定义如下, $x[n]$ 是输入信号, $h[n-m]$ 是平移后的脉冲响应。

$$y[n] = x[n] * h[n] = \sum_{m=-\infty}^{m=+\infty} x[m] \cdot h[n - m]$$

(11.1)

卷积存在许多有趣的特征,其中包括交换律,即把输入信号看做脉冲并把脉冲响应看做输入信号时卷积结果不变,反之亦然。另一个有趣的特征是,时域上的卷积等于频域上的乘积,如式 11.2。如果式 11.2 恒成立,则卷积的交换律可由式 11.2 的右边部分得证,因为乘法本身的乘积因子是可以交换的。

$$y[n] = x[n] * h[n] \leftrightarrow DFT\{y[n]\} = DFT\{x[n]\} \cdot DFT\{h[n]\}$$

(11.2)

$$y[n] = IDFT\{DFT_x \cdot DFT_h\}$$

(11.3)

因此,卷积的输出 $y[n]$ 可以通过分别计算 $x[n]$ 和 $h[n]$ 的 DFT,二者相乘后再进行逆变换获得。这样计算卷积的主要原因之一是,相较于式 11.1 的时域卷积过程,用 FFT 来计算式 11.2 和 11.3 中的 DFT 变换与 IDFT 变换的速度更快。大体上说,基于 FFT 的卷积算法比时域卷积算

法效果更好。式 11.2 的证明并不困难,而且由进一步观察可知该式相当直观(有兴趣的读者也可将该证明作为周日慵懒的午后在当地咖啡馆中休闲时的作业)。例如,将大厅的脉冲响应与一个录制好的干声信号进行卷积,就可获得“在该大厅中”聆听该声音的听觉效果。从频域的角度看,卷积就像是信号的频谱相互被另一信号的频谱因相乘而产生了声染色——乘法在布尔代数(Boolean algebraic)中的意义是“和(AND)”运算,当然在某些情况下,乘法也包含“或(OR)”运算。

循环卷积与时域混叠

以上所讨论的卷积方法实际上称为“循环卷积”(circular convolution),并带有人为失真。循环卷积的人为失真将使卷积后的信号发生时域混叠(time-alias)。接下来举例说明什么是时域混叠。令输入信号 $x[n]$ 的长度 $M=512$, 系统的脉冲响应 $h[n]$ 长度 $L=500$ 。由第 5 章第 5 节可知,使用时域卷积算法所得的卷积信号 $y[n]$ 的长度为 $512 + 500 - 1 = 1011$, 即为 $M + L - 1$ 。如果使用循环卷积算法,在频域中求解 DFT 变换时,可能会默认使用更大的频率窗,因此会对信号进行零填充。也就是说,如果 DFT 变换长度 $M=512$ (实际也可使用 FFT 算法,不过计算结果相同),则将在系统的脉冲响应信号尾部填充 12 个零点以使用于卷积的两个信号具有相同长度。使用 FFT 计算 DFT 时,情况也是如此。然后,在频域进行乘法,最后再对 X^d 和 H^d 进行 IDFT 变换得到 $y[n]$ 。此时会看到,经循环卷积所得的信号长度只有 512,而不是 1011 ($M + L - 1$)。这种现象就称为时域混叠。

图 11.1 给出了一个长度 $M=146$ 的输入信号(图 11.1 的上图)与长度 $L=73$ 的脉冲响应(图 11.1 的中图)基于 DFT 进行卷积的正确输出结果(图 11.1 的下图)。此例因 DFT 变换的长度太小而发生时域混叠,正确卷积后信号尾部对应的 72 个采样点($218 - 146 = 72$)溢出至信号的前部,如图 11.2,上、中、下三图给出了同一卷积后信号不同部分的子图像。由图可知,时域混叠信号只有 146 个采样点,而正确的卷积后信号有 218 个采样点。该信号尾部 72 个采样点的溢出量叠加至信号前部时会使输出产生失真。净时域混叠的输出信号如图 11.3。

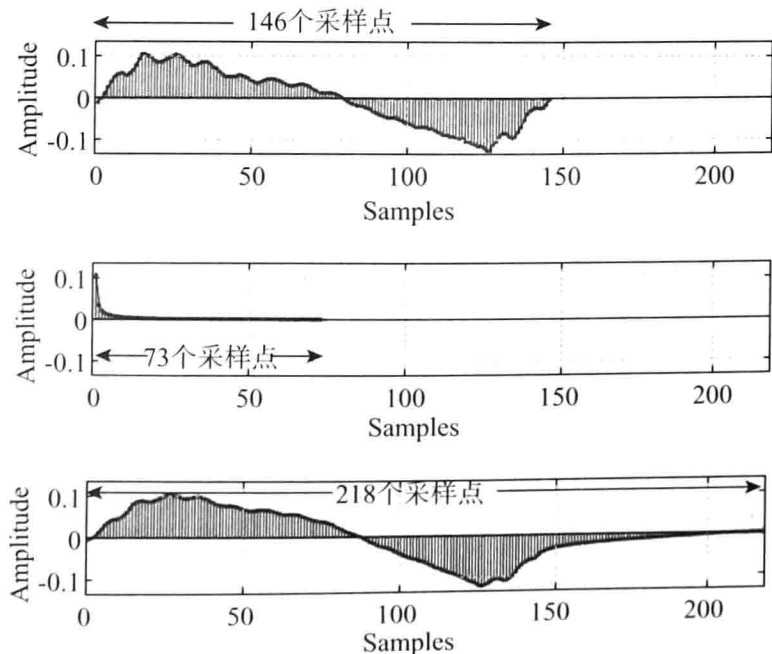


图 11.1 输入信号(上图),脉冲(中图),卷积输出结果(下图)

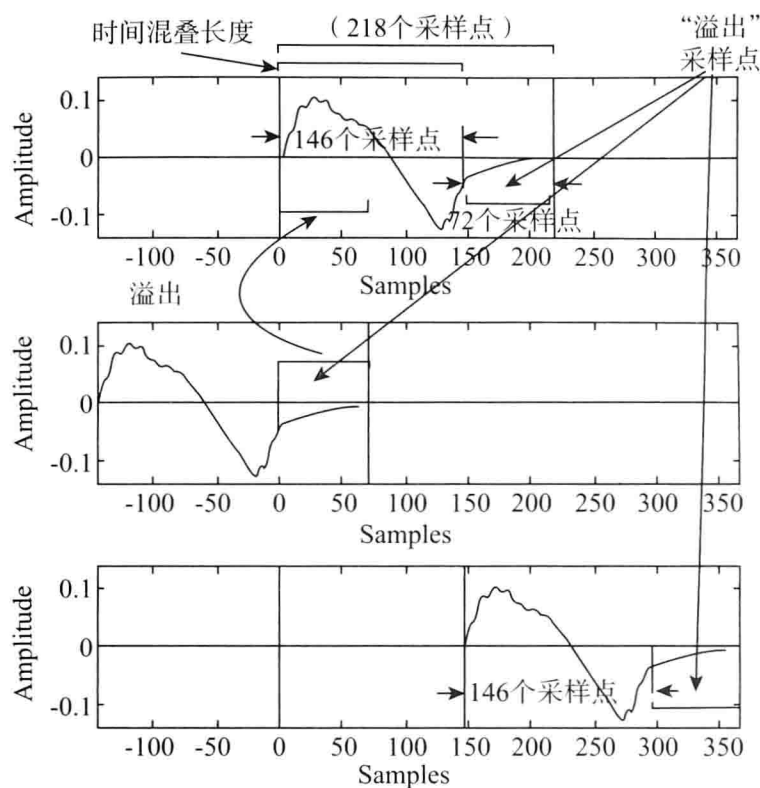


图 11.2 循环卷积的时域混叠

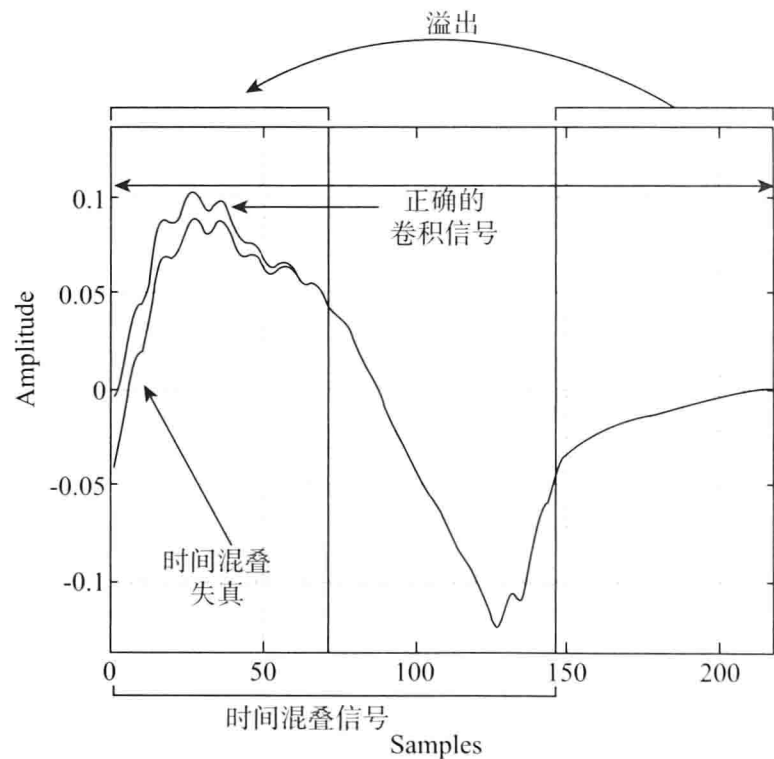


图 11.3 循环卷积时域混叠的网络结果

将输入信号零填充至正确的输出信号长度 $M + L - 1$, 即 DFT 变换长度, 可避免发生时域混叠。使用 FFT 算法时, 为了提高计算效率, 变换长度应当为 2 的整数次幂。因此, 在选择 FFT 大小时, 只需选择与输入信号长度最为接近的、较大的一个 2 的整数次幂即可, 然后按该长度在循环卷积前对输入信号进行零填充, 再从 IFFT 变换后的输出信号尾部删除多余采样, 以满足

$M+L-1$ 的长度要求,如图 11.4:

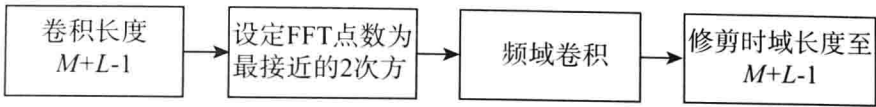


图 11.4 基于 FFT 的频域卷积过程

卷积的逆过程称为“解卷积”(deconvolution)。换句话说,如果已知系统的脉冲响应和输出信号,想知道输入信号是什么,那么有没有方法可以得到输入信号呢?回答是肯定的——即进行解卷积。在时域上处理反卷积非常困难,而幸运的是在频域上很简单,它变成了未知 DFT 帧的除法。也就是说,因为在频域上,卷积是输入信号和脉冲响应的频率帧的乘积(式 11.4),所以如果其中任何两个分量已知,就可计算第 3 个量(3 个量分别为输出信号、输入信号与脉冲响应的 DFT 变换)。

$$DFT_y = DFT_x \cdot DFT_h \tag{11.4}$$

例如,如果已知系统的脉冲响应和输出信号,可通过 $h[\cdot]$ 和 $y[\cdot]$ 的 DFT 变换的除法计算 $x[\cdot]$ 的 DFT 变换结果,如式 11.5:

$$DFT_x = \frac{DFT_h}{DFT_y} \tag{11.5}$$

12. 抖 动

早在第 1 章 7.1 节,我们就讨论了抖动,以及量化和取整所产生的谐波失真。这一现象可由傅立叶变换证明,同时,如本章一开始讨论加法合成时所述,方波可由基频及其奇次谐波描述。接下来让我们从另外一个角度来讨论第 1 章如图 12.1 所示的抖动处理问题。

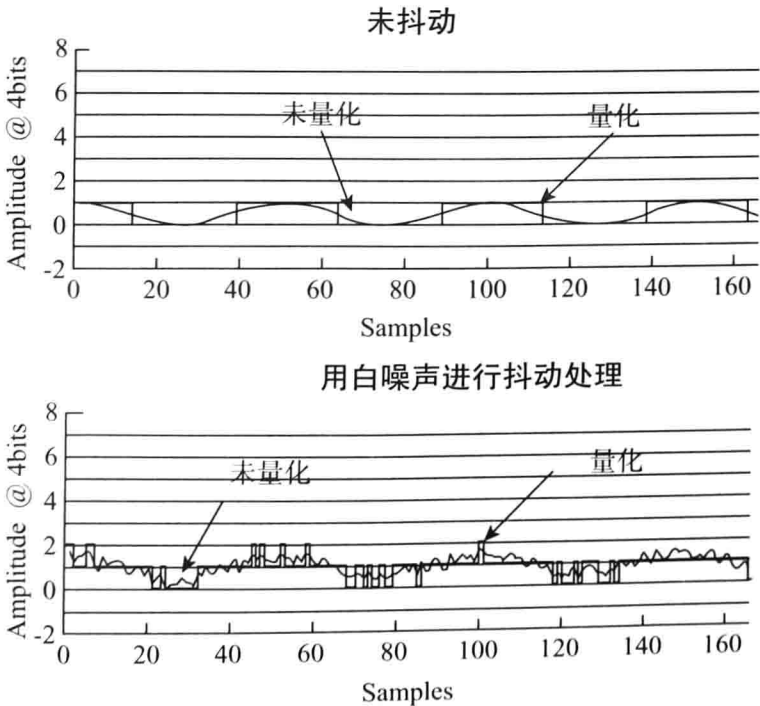


图 12.1 经白噪声做抖动处理的低幅值余弦波

图 12.1 的上图给出了对正弦波进行 4 bit 量化和采样时所发生的情况,下图给出了用白噪声进行抖动处理前后的量化结果。虽然原始模拟信号是平滑的,但未经抖动处理的量化信号看起来更像一个方波。图 12.2 为上述两个量化信号的傅立叶变换,由图可知,上图有谐波失真——这些谐波并不是所要的量化结果,数字化输出的取整过程产生了失真。下图为抖动处理后的傅立叶变换结果,只有一个对应正弦波频率的分量(即只有一个突出峰值),在低能量本底噪声上没有其他谐波分量。如此低的本底噪声在大多数情况下都无法被感觉到,特别是当信噪比很高的时候,信号的有效声级将掩蔽背景噪声。这类现象也非常直观——由图可知,我们也许立刻注意到的是这个突出峰值,而不是看起来像草地一样的本底噪声。所以,该突出峰值的背景看上去像被蒙上了一层阴影。

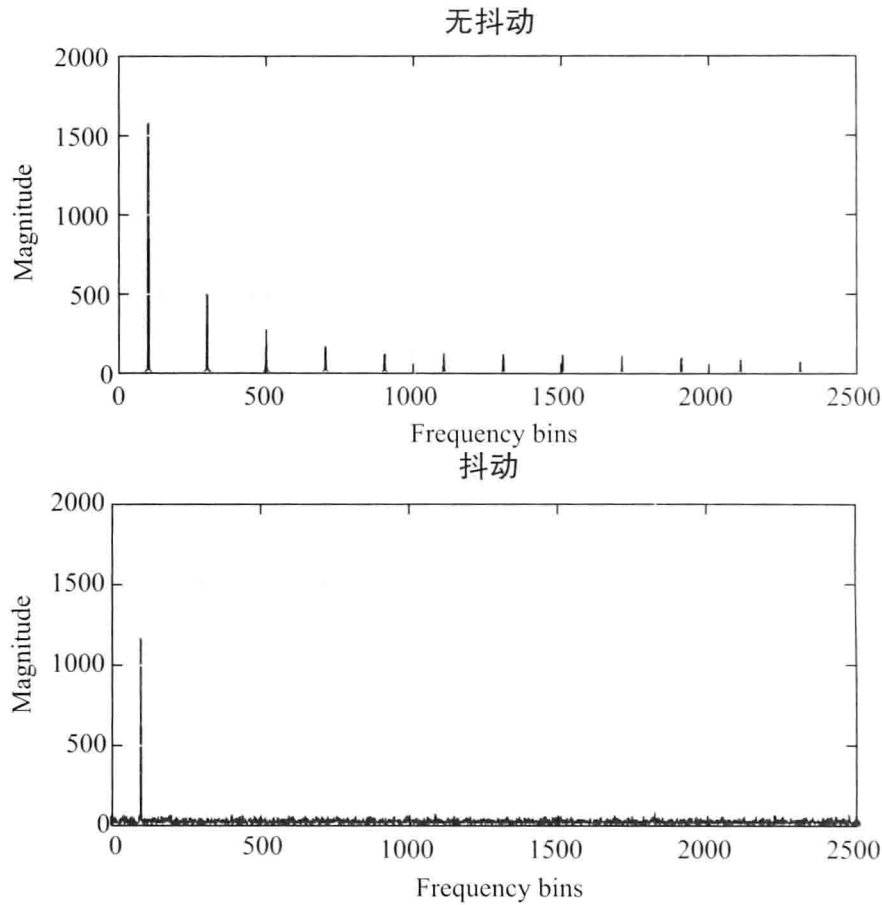


图 12.2 抖动处理前后的上述余弦波的傅立叶变换

13. 光 谱 图

虽然单个 DFT 变换的频率帧已经能反映信号的频率结构,但尤其对于实际信号——如音乐和作曲中每天使用的音频信号,得到的和可提取的有用信息并不多。那些不随时间变化的信号——即稳态信号或许会使音乐作品变得乏味,当然,除非该信号在一分钟内有微小变化或只是持续的嗡嗡声。不过即使是在动态变化很小的作品中,声音的大小多少也会有所变化,虽然这些变化的声音不会被认为是前景音乐。本章第 5 节提到的短时傅立叶变换对分析这类声音很有帮助——它可在保持频率分辨率鲁棒性的同时,分析瞬态声音频谱的每一帧。光谱图 (Spectrogram) 就是一种基于多向量的 STFT 变换,它在三维空间上进行映射,平面上的两维度分

别为时间和频率,第 3 个维度为幅值,常用色彩编码(或灰度系统)表示。为了实际操作方便,通常用 x 轴表示时间,用 y 轴表示频率,如图 13.1:

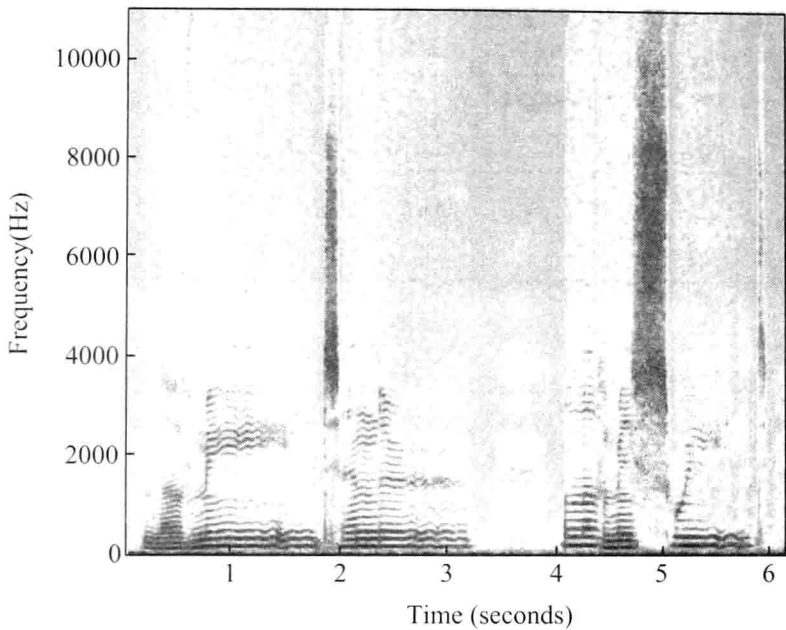


图 13.1 光谱图

光谱图第一眼看起来有些难于理解(虽然它看起来总是很酷)。不过,通过光谱图(增强型的傅立叶)有助于对信号进行视觉化分析,该图的水平分量(周期或准周期)为时域信号,垂直分量为频率信号,与傅立叶变换中的频率轴相对应。如果是持续的周期(也适用于一段时间),我们将看到一个水平线上形成的光谱图。如果信号具有周期性(在某个时域内),则一般会在光谱图上看到一条颜色或灰度周期变化的水平线。如果信号在时域上(水平方向)是非周期的,DFT 帧的频谱平直,在光谱图上会出现一个噪点块——或者在垂直方向上的谱线周期变化较少。在光谱图上有了这些关于信号的基本属性,由图可知,该信号约在 2 秒前一小段、3 至 4 秒之间、5 秒附近和光谱图尾部是非模式化(非周期)的类噪信号,在其他区域是具有谐波的有调声,偶尔还可观察到信号经过轻微的低频调制和振幅调制。该信号实际是男声演唱“love me tender, love me sweet”的声音。一般来说,演唱或语音中的爆破音会产生噪声,如该信号 2 秒左右约是单词“tender”中的“t”的发音位置,5 秒处约是单词“sweet”中的“s”的发音位置。我们将在下一章讨论语音声码器和提取 LPC 滤波器,该滤波器通过修改和提取 LPC 系数可用于分析辅音和元音的不变特征。

14. 傅立叶变换性质小结

本节总结了傅立叶变换中的一些重要性质,如表 14.1 所示。其中一些性质比其他更明显,不过无一例外的是,所有这些性质都有助于处理傅立叶变换。表 14.2 则总结了不同类型傅立叶变换的性质。

表 14.1 傅立叶变换的重要性质

性 质	描 述
线性 (Linearity)	$g[n] = ax[n] + by[n] \leftrightarrow Z(\theta) = aX^f(\theta) + bY^f(\theta)$ $a, b \text{ 为常数} \quad (14.1)$
周期性 (Periodicity)	$X^f(\theta) = X^f(\theta + 2k\pi)$ $\theta \text{ 为实数}, k \text{ 为整数} \quad (14.2)$
时移性 (Time shift)	$y[n] = x[n - m] \leftrightarrow Y^f(\theta) = e^{-j\theta m} X^f(\theta)$ $m \text{ 为整数} \quad (14.3)$
频移性 (Frequency shift)	$y[n] = e^{-jkn} x[n] \leftrightarrow Y^f(\theta) = X^f(\theta - K)$ $\kappa \text{ 为整数} \quad (14.4)$
乘法律 (Multiplication)	$y[n] = x[n] \cdot h[n] \leftrightarrow Y^f(\theta) = X^f(\theta) * H^f(\theta) \quad (14.5)$ $y[n] = x[n] * h[n] \leftrightarrow Y^f(\theta) = X^f(\theta) \cdot H^f(\theta) \quad (14.6)$
对称性 (Symmetry)	$X^f(-\theta) = \widehat{X^f(\theta)} \quad (14.7)$ $\text{Re}\{X^f(-\theta)\} = \text{Re}\{X^f(\theta)\} \quad (14.8)$ $\text{Im}\{X^f(-\theta)\} = -\text{Im}\{X^f(\theta)\} \quad (14.9)$ $ X^f(-\theta) = X^f(\theta) \quad (14.10)$ $\angle X^f(-\theta) = \angle -X^f(\theta) \quad (14.11)$

表 14.2 不同类型的傅立叶变换小结

时域 频域	连 续	离 散
连 续	时间连续傅立叶变换： $X^f(\omega) = \int_{-\infty}^{+\infty} x(t) e^{-j\omega t} dt$ $x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} X(\omega) e^{j\omega t} d\omega$	时间离散傅立叶变换： $X^f(\theta) = \sum_{n=-\infty}^{n=+\infty} x[n] e^{-j\theta n}$ $x[n] = \frac{1}{2\pi} \int_{-\pi}^{+\pi} X^f(\theta) e^{j\theta n} d\theta$
离 散	时间连续傅立叶级数： $x(t) = x(t + T)$ $X^s[k] = \frac{1}{T} \int_0^T x(t) e^{-j\frac{2k\pi t}{T}} dt$ $x(t) = \sum_{k=-\infty}^{k=+\infty} X^s[k] e^{j\frac{2k\pi t}{T}}$	时间离散傅立叶变换： $X^d[k] = \sum_{n=0}^{n=N-1} x[n] e^{-j\frac{2k\pi n}{N}}$ $0 \leq k \leq N - 1$ $x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X^s[k] e^{j\frac{2k\pi n}{N}}$

15. MATLAB® 与傅立叶变换

MATLAB® 的“数字信号处理工具箱”(Digital Signal Processing Toolbox)包含了 FFT 变换函数和用于生成光谱图的 STFT 变换函数,以及其他一些与傅立叶变换有关的函数。这些函数的使用都非常简单,至少 STFT 和 FFT 变换函数都是直接可以使用的。代码 14.1 是一个对 4 kHz 正弦波进行 FFT 变换的例子,变换后的幅值分别用线性、对数和分贝坐标表示。

```

f = 4000;
fs = 22050;
fftLength = 1024;           % window length
x      = sin(2*pi*f*[0:1/fs:1]); % make the sine wave
ft     = fft(x, fftLength);   % do FFT, use rect. window
ftMag  = abs(ft);             % compute magnitude

% plot the results both in linear and dB magnitudes
subplot(2, 1, 1), plot(ftMag)
title('Linear Magnitude')
ylabel('magnitude'), xlabel('bins')

subplot(2, 1, 2), plot(20*log10(ftMag))
title('dB Magnitude')
ylabel('dB'), xlabel('bins')

```

代码 14.1 快速傅立叶变换 FFT

16. 乐例分析

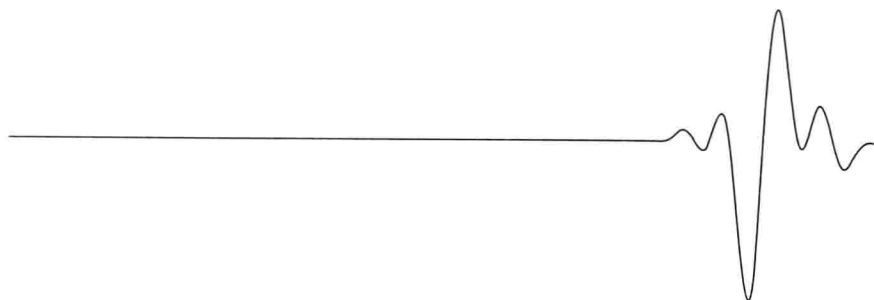
可以毫不夸张地说,傅立叶变换在如今的音乐世界中无处不在,尤其是插件众多的数字音频工作站 DAW (digital audio workstation) 更是离不开它。使用计算机工作的大多数作曲家已经在积极地使用(不管是否知情)基于傅立叶变换的软件来进行作曲、声音编辑、混音、母带处理以及其他音乐处理工作。显然,使用傅立叶变换没有什么特别的,与此相反,正是因为该技术的使用如此平常和普及,没有人会真正去谈论与它有关的话题。不过,在作曲家还没有使用计算机去琢磨声音的特定频谱结构的时代,就已经有很多音乐作品借用了傅立叶变换的思想。在 20 世纪 50 年代,作曲家在此方面进行了大量的研究工作,尤其在德国,如施托克豪森这样非常关注(早期)电子音乐(Elektronische Musik)而不是具像音乐(Musique Concrète)的作曲家。那时候,电子音乐仅是由振荡器的声音组成的音乐,而具像音乐技术则以声音作品为中心来“发现声音事件”(found sound objects)——这些现实声音由麦克风和磁带录音机录制。施托克豪森在 1952 年创作的作品《声音研究 I》(Klangstudie I)中使用了加法合成,其中对两个振荡器输出信号的层叠是一种乏味繁琐但时髦的做法——即同时播放两个振荡器的输出信号并把它们录制在同一个磁带机上,然后用同样的振荡器再播放另外一对纯音信号,同时再用同样的磁带机播放之前录好的正弦波,最后将这些信号叠加后的波形录制在另一台磁带机上。这些研究像是基于物理学家赫尔曼·冯·亥姆霍兹(Herman von Helmholtz)对音色的研究。亥姆霍兹曾使用被称为亥姆霍兹共振器(Helmholtz resonator)的装置,通过校准好的长颈瓶来实现傅立叶的手动分析。这些亥姆霍兹共振器有两个开口,一个拾取外界的声音,另一个对准耳朵用于监听。每个亥姆霍兹共振器中的长颈瓶都校准为既定的频率和音调,如果通过共振器的输入信号包含有该既定频率分量,则在监听端能听到该频率的声音。最近仍然大量使用傅立叶变换进行音乐创作的作曲家是保罗·昆斯(Paul Koonce)。其中一首这样的作品是《热屋》(Hot House, 1996)。在该作品中,作曲家很好地处理了音色的非谐波失真(timbral enharmonic change)问题。也就是说,他在处理有关音色共性和从一个声音事件变化(或变形)为另一个声音事件的问题时,采用

音色特性相近的声波进行叠化。如乐曲中有一段,声音从汽笛声慢慢地变成一个火车驶过的声音。反过来,火车停下来蒸汽释放的声音则标志一个音乐动机的结束,然后过渡至背景环境声,如人群的谈话声和高呼声等,突然出现一个不稳定的声音对象,似有人从人群中穿过,最后过渡到在家里喝茶的声音。利用这些音色相近的声音的叠化,由一个声音空间变化至了另一个声音空间,由汽笛声最后转化为在家里喝茶的声音,乐曲的旅程得到了合理的推进。

参考文献与扩展阅读

- [1] Porat, B 1997. *A Course in Digital Signal Processing*. Wiley and Sons.
- [2] Cooley, J. and J. Tukey 1965. "An Algorithm for the Machine Calculation of Complex Fourier Series", *Mathematics of Computation*, 297 – 301.

第9章 频谱分析、声码器及其他问题



1. 引言

本书的最后一章将介绍一些声音的谱分析与合成,以及在音乐应用中有趣而实用的方法,同时还将讨论一个和语音有关的有趣问题,称为声码器(vocoder)。接下来,将从一系列谱分析技术开始本章的旅程,这些技术包括峰值检测和基频检测,之后的议题再围绕着在音乐作曲、音乐制作和语音合成等领域中非常流行的声码器展开。

音乐信号与重要术语

几乎可以确定的是,相对于人类文化遗产,本书之前接触到的音乐实例可能已经超出了传统音乐的范畴,有些声音甚至听起来不像音乐,或者是介于音乐与非音乐之间。很难、甚至不可能在两者之间找到明显的界限——即何为音乐声,何为非音乐声。困难之处很大程度上在于人们的生活背景往往对声音和音乐的感知有着重要作用。对某些人来说,噪声信号也许不能被忽视(如飞行员耳中飞机螺旋桨旋转的声音,又或者是深夜在野外训练的军人听到的池塘里青蛙的叫声),同时某些人可能会认为这些噪声本身就具有音乐性,并利用这些噪音素材进行创作。一些伟大的音乐作品,包括披头士乐队(the Beatles)的《回到苏联》(*Back in the USSR*)和平克·弗洛伊德乐队(Pink Floyd)的《金钱》(*Money*),它们的声音不能称为传统意义上的“音乐”,而是有效增加歌曲体验的声音效果。从某种意义上说,平心而论,从正弦纯音到复合音,再到白噪声,声音的频谱实际上是无限的。而其中另一些声音的频谱却存在特定的规律和模式,或者说是有调的,并有可能被作为乐音而得到大众接受(至少现在可以这样说)。本书所涉及的大多数声音都是这样的有调性声音——或者是准和谐的有调声。

有调声是有规律的——用来描述其频域频谱基本结构的术语包括峰值(peaks)、谐波(harmonics)、分音(partials)和泛音(overtones)。这些术语在很多方面尽管比较相似,甚至有时会被误用,但它们的概念却十分具体。在这些术语中,“峰值”一词大概最为常用,它用于表示频谱的突起,指频谱中任何一个正向斜面与紧接着的负向斜面所构成的突出点。峰值通常突出于背景噪声,有可能是一些幅值较小的峰值或底噪本身的一部分,在这种情况下,尽管从技术角度看这些突起也是峰值,但这些值也许并不能称为峰值。频率轴上的峰值位置并不一定有规律可

循。那些与基频不成整数比的峰值称为分音——因此,那些组成复合音的、幅值较强的突出频率分量之间不一定是整数比的关系。而那些互相成整数比(或近似整数比)的峰值称为谐波,这些峰值往往都是基频的整数倍——而基频则决定了声音的音高,如式 1.1 所示, n 为整数。

$$f_n = n \cdot f_0 \quad (1.1)$$

一次谐波通常称为基频,用特殊符号 f_0 表示,二次谐波($n=2$)比基频高一个八度,三次谐波频率则是基频的 3 倍,依次类推。泛音一词常用于音乐学,指所有频率高于基频的谐波。需要防止混淆的是,一次泛音对应的是二次谐波,二次泛音对应的是三次谐波。当复合音只包含奇次(或偶次)谐波时,这种术语上的混淆问题更加严重。对于只包含奇次谐波的复合音,如果说二次泛音实际指的是基频的第一个奇次谐波即三次谐波,而二次泛音实际上是指五次谐波。因此需要注意的是,在正确的场合应当使用正确的专业术语。

有调声的音色(timbre)特征,或声音色彩(sound color),又或其他相近的特征受谐波结构的影响很大——包括谐波的位置和谐波的幅值。这一受影响的音色范围从具有代表性的单簧管扩展到弦乐器,单簧管的音色特点是奇次谐波能量较强,而弦乐器的谐波则轻微展开(靠近奈奎斯特频率的理想位置时,谐波间距逐渐增大)。大多数情况下,真正具有式 1.1 的谐波结构的声学乐器和音乐场合几乎没有,真实的谐波通常位于接近理想谐波的位置。实际谐波结构与理想结构之间的误差也许是音色中最令人感兴趣的因素之一,它能使声音变得“生动”而“自然”。由于音色问题本身需要大量篇幅来进行论述,如果不是一整本书的话可能无法深入分析,因此本书在此也不再深入讨论。下一节将回过头来介绍一些基本的谱分析技术。

2. 频谱分析

频谱分析指在频域对信号的特征进行考察,最常用的方法是傅立叶变换。使用计算机进行分析时,则一般通过上一章介绍的 DFT 和 STFT 实现。直言不讳地说,进行频谱分析就是在频域寻找信号的特征模式、特征参量和特征值,为获得既有的信号特性提供线索。本节的议题与内容将围绕着频谱分析展开。

2.1 长时平均频谱

长时平均频谱 LTAS(long-term average spectrum)的计算如式 2.1,它用于考察声音事件(或任何信号)的平均频谱。该基本算法的优势在于,背景噪声可被均值计算消除,频谱上的峰值将更加明确而且突出。产生这一结果的原因是,如果背景噪声平稳,特别是幅值与均值偏差不大,则均值算法在本质上可消除背景噪声并增强峰值。

$$\text{LTAS} = \frac{1}{J} \sum_{m=0}^{M-1} X_m^d \quad (2.1)$$

M 是 STFT 帧的数量, m 是帧序号, X_m^d 是包含幅值和相位第 m 个频谱帧。同时,进行 LTAS 分析时(由于均值计算通过低通滤波器实现)会损失一些瞬态信息。因此,按使用场合和需要,LTAS 适用于在进行更细致的分析之前对信号进行全局预分析。另外,在某些不需要精确结果的场合,LTAS 可能是一个完美的解决方案。图 2.1 是一个采样率 $f_s = 44.1 \text{ kHz}$ 的电贝斯波形。图 2.2 是该电贝斯信号中段的 DFT 变换结果与 LTAS 频谱,窗宽为 1,024 点,窗重叠率为 40%。

图 2.2 的上图是前 139 帧的 DFT 变换结果,下图是 LTAS 频谱。在这种情况下由两图对比可知,LTAS 能更好地获得反映该例谐波与基频整体结构的频谱图像。

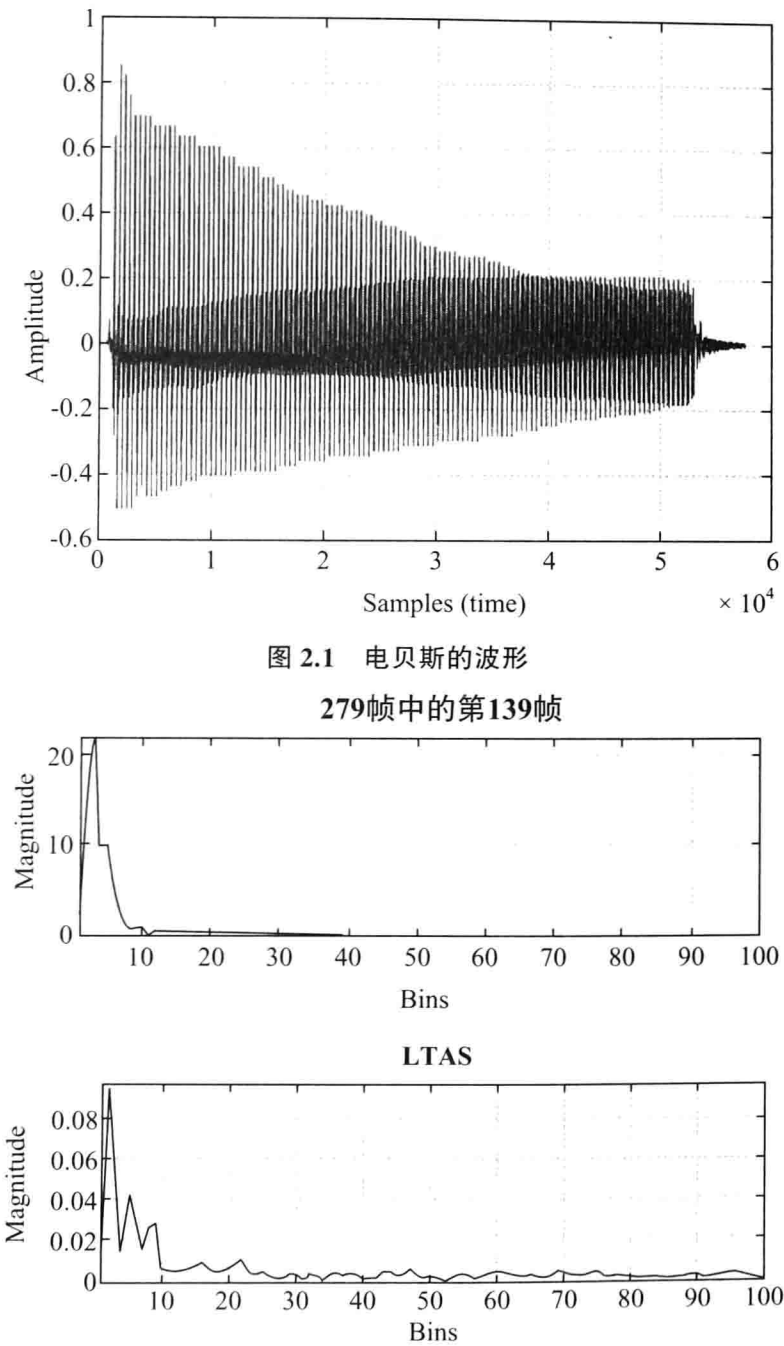


图 2.2 电贝斯的 DFT 和 LTAS 变换

2.2 对数坐标与线性坐标

分析时窗化信号的频谱时,通常有很多种观察和分析结果的方法。其中一个选择是用以 10 为底的对数坐标,而不是线性坐标来表示幅值。两者的主要区别在于,对数坐标一般能提供更多幅值细节,第 6 章讨论的线性坐标的细节则较少。提到从对数幅值曲线能获得更多信息,可能会有些误导,因为对数坐标并没有增加额外的信息。但由于对数函数的特征,如图 2.3,该图像能使较大的值变得较小,同时使较小的值变得较大,因此显现出了那些通常不易观察到的较小幅值。图 2.4 和图 2.5 给出了前例电贝斯频谱的线性幅值和对数幅值曲线。

而线性幅值曲线并非没有用处,实际上,它与对数幅值曲线同样重要。由图 2.4 和图 2.5 可知,两者对于表现较强的突出峰值和谐波共振结构都特别有用。选择对数坐标还是线性坐标要视所需要的具体情况而定。不存在哪一种方法更好,因此它们仅仅是两种不同的表示方法而已。

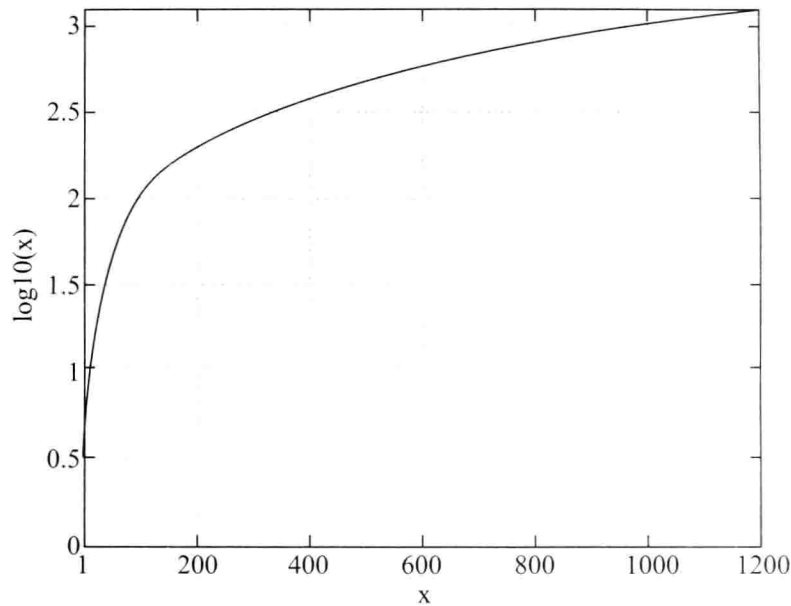


图 2.3 对数函数

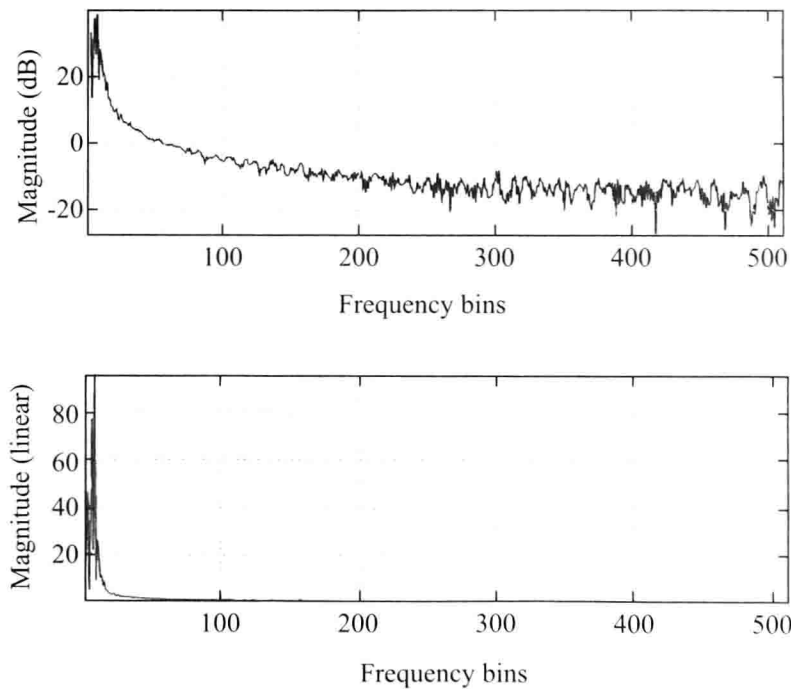


图 2.4 全频域的对数谱与线性谱

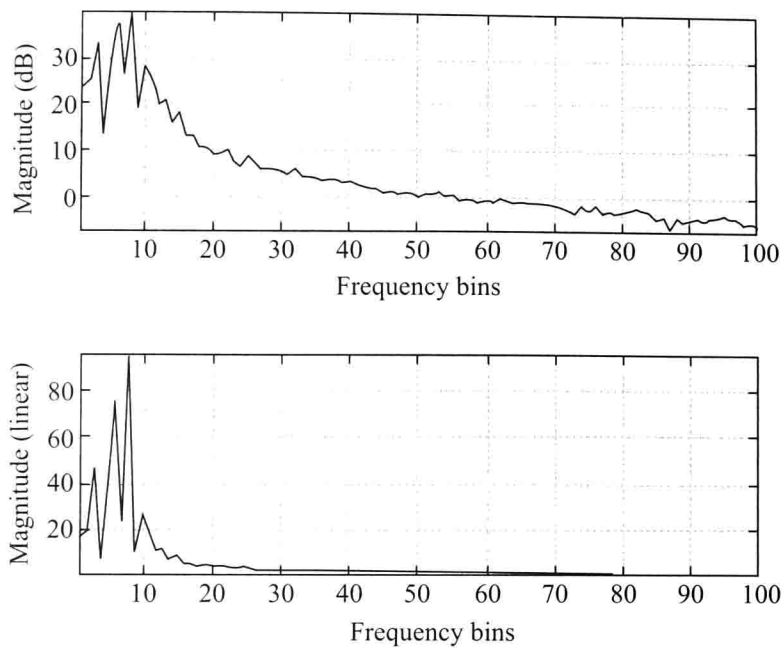


图 2.5 前 100 个频点的对数谱与线性谱

对数坐标也常用于 x 频率轴。例如,由于人耳对频率的感知是对数非线性的而不是线性的,因此对数频率坐标有时与人耳处理有调声信号时的声音感知特性有关。“八度”(octave)是理解音高非线性特征的简单例子,其定义是基频的倍频(或参考频率的倍频)。如基频 $f_0 = 100\text{Hz}$,第 1 个八度将出现在 200Hz 。下一个八度将出现在 400Hz ($100 \cdot 4$ 或 $200 \cdot 2$,即二次谐波的八度), 400Hz 之后的八度则将出现在 800Hz ,依此类推。人耳听觉系统所感知到的八度递增规律为——频率增加至原频率的两倍。而实际上,频率增量频率也根本不是线性的。对于上例,设可容许的频率分辨率为整数值,则第 1 个八度 f_2 与基频 f_0 的“频率距离”(frequency distance)是 100Hz ,第 2 个八度 f_4 和 f_2 的距离是 200Hz ,第 3 个八度 f_8 和 f_4 的距离将是 400Hz ,依此类推。虽然对于每个八度的平移,人耳感知到的频率变化,但实际频率距离却在频率轴上扩大了。使用对数坐标可以更好地体现这种感知上的线性特征,虽然这不是数学意义上的线性。如果仍然难于理解,再以钢琴键盘为例:钢琴的琴键都有相同的物理间距——所有白键具有相同的特定宽度,而所有黑键也有另一相同的宽度。然而,尽管频率距离随八度递增,但物理键盘的琴键距离不变——这便称为线性。当我们弹奏钢琴时也能感觉到这种布局的正确性,即钢琴键盘的刻度并不会让演奏者感到不便。而弦乐器的情况相当于钢琴琴键的距离发生变化——即对于弦乐器,演奏的频率越高,品之间的距离越短。这种情况当然取决于频率增加的程度——从吉他的琴颈开始升高音阶频率,则频率的扩展导致波长的降低,从而使品与品之间的距离变短(波长与频率成反比)。

2.3 频谱的峰、谷和频谱包络

由频谱图 2.6 可获得一些有用的特征参数,其中两个重要的特征是频谱的“峰”(peak)和“谷”(valley)。频谱的峰非常重要,它能反映声音事件的谐波共振结构。频谱的谷可看成阻尼或反共振点,从滤波器的角度看,对应着系统的零点——可以想像极点位于峰值处,零点位于峰值之间。

频谱的峰和谷决定了声音事件的总体音质。而频谱中也存在一些细微的、也许同时相对于峰和谷的主导作用来说不那么重要的参数。在某种程度上,观察频谱时,第一眼所看到的是由幅值

走向决定的频谱轮廓、外观和形状,称为“频谱包络”(spectral envelope)。我们(人类)直接观察频谱的突出峰值(或谷)非常简单,但让计算机自动识别这些峰谷特征则比较困难。对于问题似乎都存在这样的情况——人类相对容易解决的问题计算机很难解决,而计算机容易解决的问题(如,由0开始求前两百万个奇数之和,再求其平方根,结果再除以5.100276)人类却很难解决。

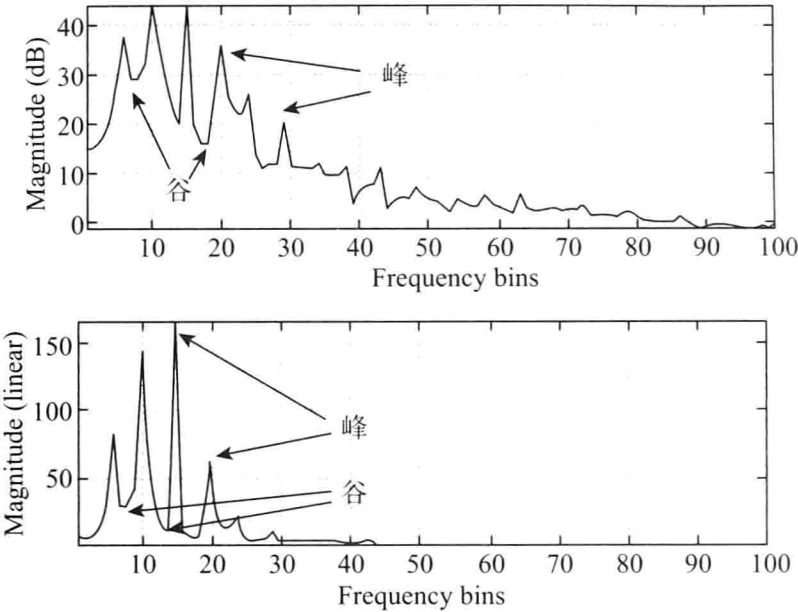


图 2.6 频谱中的峰值和谷值

如本章一开始所述,峰的定义非常直观,为特定斜率变化的拐点——由正向至负向。谷则恰恰相反,由负向至正向的斜率变化拐点。而这种分析有时并非完全有用,因为分析结果的数据量往往非常巨大,如图 2.7 所示:

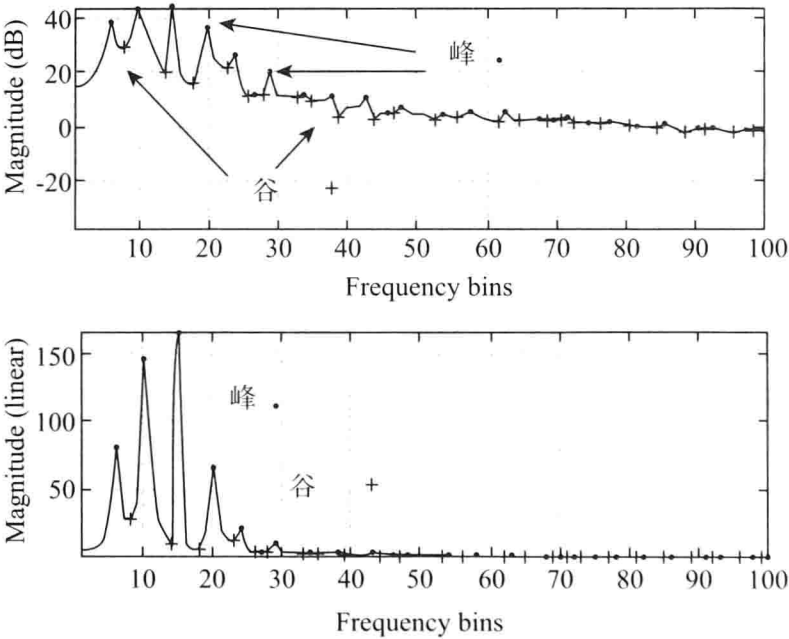


图 2.7 频谱中的全部峰和谷

图 2.7 仅沿频谱计算了斜率极性的变化,并将峰定义为斜率由正向变为负向的拐点,将谷定义为斜率由负向变为正向的拐点。此外,可利用峰值分析器,将频谱垂直翻转,再通过寻找斜率由正向变为负向的拐点来分析频谱的谷。获得频谱中有用数据的方法还有很多,包括设置一

个阈值为低于最大峰值 50% 的滤波器对频谱进行滤波,或设置一个阈值,当峰值与其相邻谷值的差大于该设定值或适应值时为有效峰值,又或者仅仅提取前 10 个最大峰值和前 10 个最小谷值做为该频谱最具代表性的峰和谷。显然,所有这些方法都有自身的缺陷,由于信号的巨大信息量,实际上没有一个完美的算法能准确找出复杂频谱中所有的峰和谷。

计算频谱包络的方法也有很多,包括频谱帧的低通滤波法,或计算时域包络使用的 RMS 时窗分析法。RMS 频谱包络如图 2.8 所示。注意到使用 RMS 计算包络时,因使用时窗发生了减采样——表示频谱的数据点将会减少。因此,在 RMS 算法之后需要对频谱进行插值以使 RMS 频谱与原始谱的长度一致。

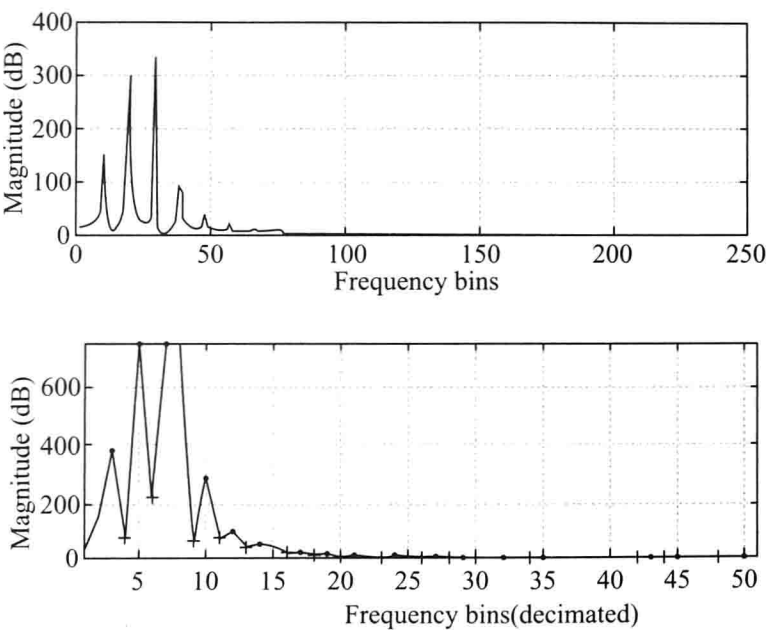


图 2.8 全频谱的峰值提取

另一种提取突出峰值的方法是短峰值 (passing peak) 滤波法——即突出峰值的集合不包括这些幅值较小的峰。不过判断哪些是“短峰值”比较困难,说起来比实际做起来要容易。图 2.9

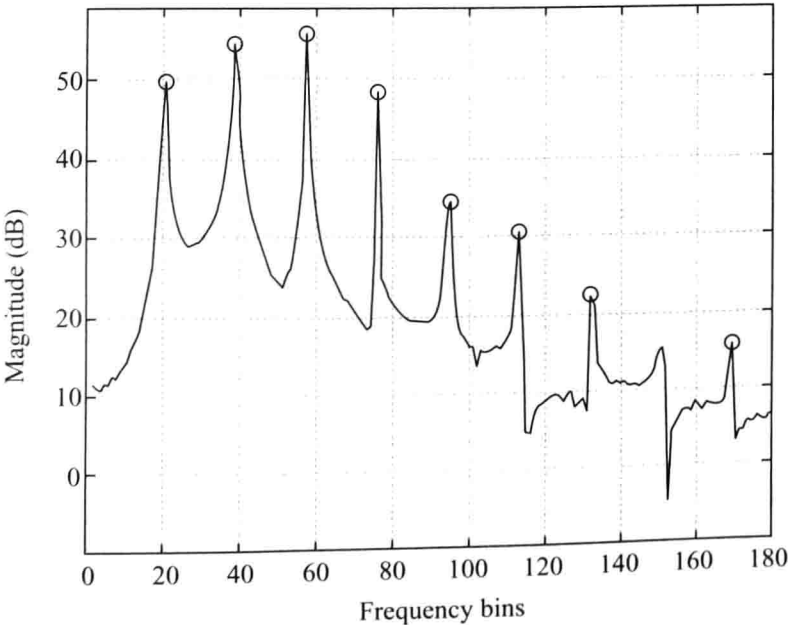


图 2.9 RMS 谱包络的峰值

给出了这一算法的结果。对于突出峰值的提取,只能做到更好,而不能做到最好——无论最好指的是什么。更多关于峰值提取算法的信息和资料可在互联网上找到,并可在网址 <http://music.princeton.edu/~park> 中下载。

2.4 基频及其谐波的提取

有调声的频谱结构通常包括基频及其谐波,其中基频是正常感知到的音高。图 2.10 显示了一个基频为 100Hz 的频谱结构。

不过并不是所有声音的基频都有最强峰值,有时候基频甚至不存在[称为“基频缺失”(missing fundamental)现象]。对于基频缺失的声音,人耳仍能通过谐波结构感知到基频所决定的音高,尽管于物理意义上,它在频谱中并不存在。此外,声音信号的音乐学音高是一个感知心理量,而频率则是一个绝对物理量,如它可用示波器测量。现实世界的乐音都是多个谐波组成的复合信号,因此也许并不奇怪的是,随着声音中谐波数量的增加,音高的感知也会增强(Truax 1978)。虽然仍有许多与音高或音调感知相关的研究与讨论,这也是一个相当有趣和非常重要的专题,但本书将回到讨论基频检测的 DSP 算法上,不会在音高感知方面做过多的论述。

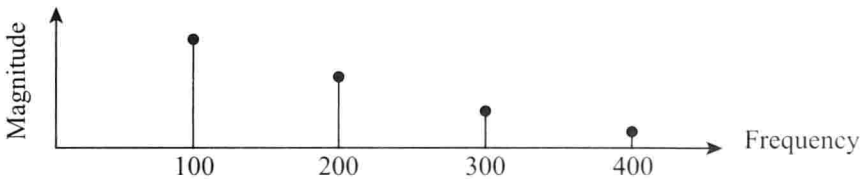


图 2.10 调性谱的谐波结构

计算基频的方法有很多,因为一种算法适用于某一种类型的信号而对于其他类型的信号效果欠佳,所以很难说哪一种方法是最优算法。第 2 章第 5 节已经讨论了一些基于时域的基频估计算法,其他的基频估计算法包括倒谱分析法(cepstral analysis),谐波谱生成法(harmonic product spectrum)和反梳状滤波器法(inverse comb-filter)。最后一种方法虽然不属于频域算法的范畴,但也是一种用反梳状滤波器的频率响应来估计基频的方法。

2.4.1 反梳状滤波法

如上所述,其实反梳状滤波本身并不是一个频域处理过程,但它在基频估计中的应用可从频域的角度理解。该滤波器的 FIR 差分方程如下:

$$y[n] = x[n] - b_1 \cdot x[n - N] \tag{2.2}$$

由第 7 章 3.3 节可知,FIR 正向梳状滤波器的差分方程如式 2.3:

$$y[n] = x[n] + b_1 \cdot x[n - L] \tag{2.3}$$

两个滤波器的形状非常相似,如图 2.11 和图 2.12,但它们的滤波效果有很大差别。由图 2.11 可知,正向梳状滤波器约在 0.1,0.3,……等频点有陷波。而反梳状滤波器则是约在 0.2,0.4,……等频点有陷波——这些频点恰好在正向滤波器的陷波频点之间。反梳状滤波器正是利用有调声具有一定谐波结构的特点来进行基频分析的。即从傅立叶变换的角度看,功率谱将显示随基频变化的强谐波,因此利用反向梳状滤波器,将其陷波位置与信号谐波位置对准后,再

对信号进行反梳状滤波,则该信号的谐波能量将大幅度衰减,基频能量将更突出。陷波频点与滤波位置对得越准,经反梳状滤波后的信号谐波能量越低。

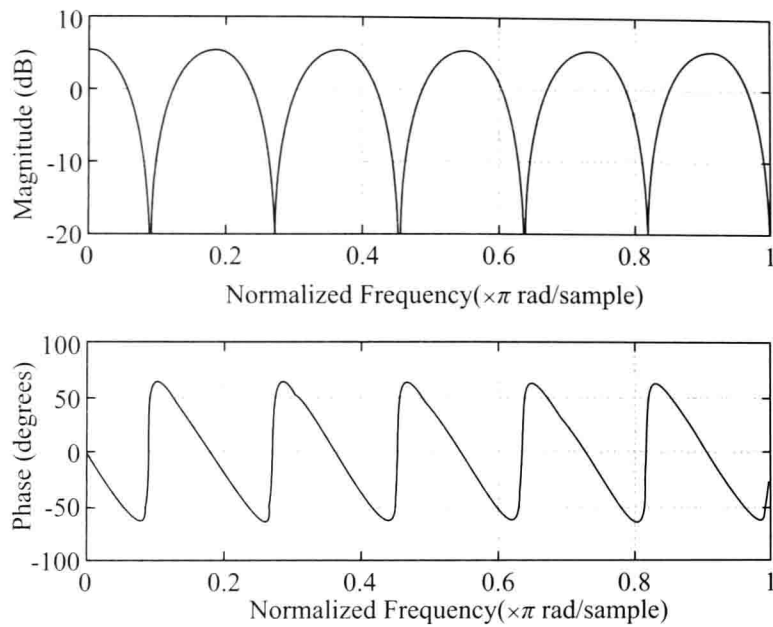


图 2.11 FIR 梳状滤波器

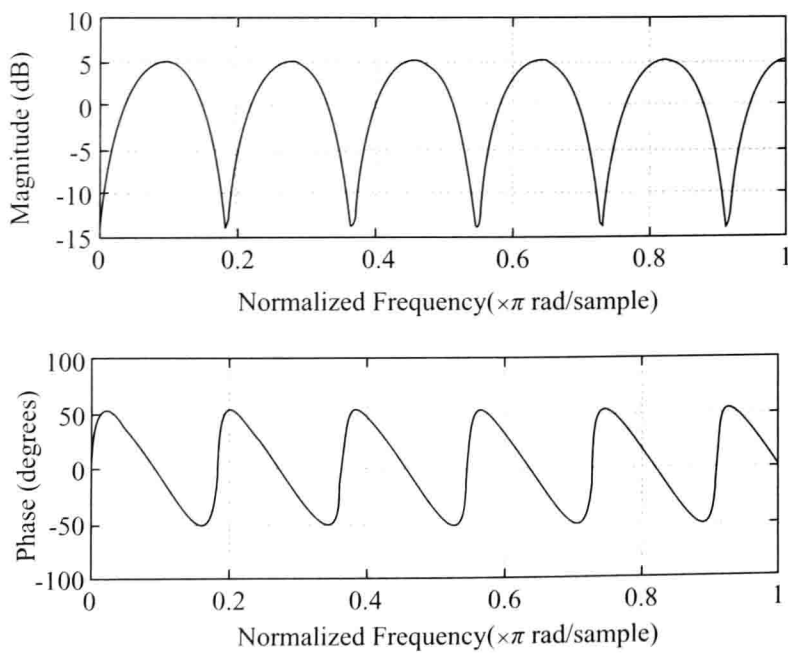


图 2.12 FIR 反梳状滤波器

图 2.13 是一个利用 IIR 反梳状滤波器提取基频的实例,滤波器的差分方程如式 2.4,输入信号为 C # 4 音高的长笛声。

$$y[n] = x[n] - a_1 \cdot y[n - N]$$

(2.4)

用反梳状滤波器进行扫频,在找到能使滤波后信号的谐波分量尽可能小的延迟量 N 之前,输入信号需要经过一个预权重(pre-emphasis)滤波器。预权重滤波器是一个非常基本的高通滤波器,往往有助于使音频信号的频谱变得平直。乐音信号通常都有较强的低频谐波分量,随着

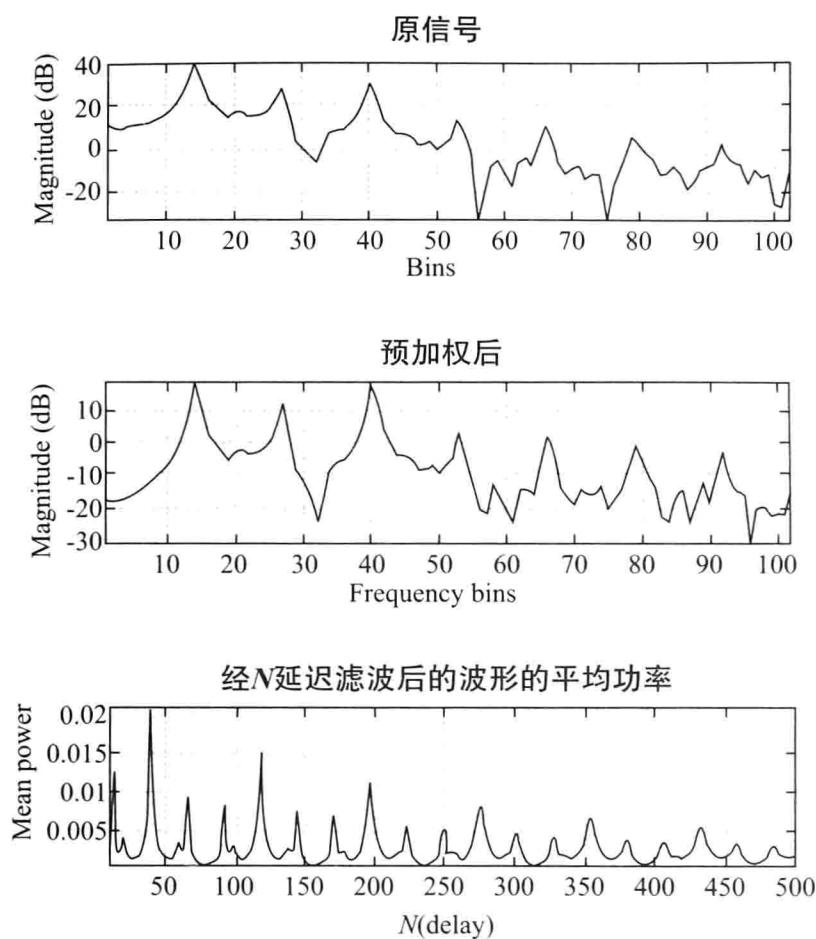


图 2.13 带预权重滤波的 IIR 反梳状滤波器,上图为原始功率谱,中图为带预权重处理的频谱,下图为经 N 阶延迟滤波的信号能量均值谱

谐波阶数的增高,谐波幅值将衰减,如图 2.10。因此,加入一个预权重滤波器,如式 2.5,对长笛乐音信号进行平直处理后,有助于利用反梳状滤波器对各谐波分量进行更均匀的陷波处理。

$$y[n] = x[n] - b_1 \cdot x[n - 1] \tag{2.5}$$

也许读者还记得,之前在第 6 章第 3 节已经介绍过这一高通滤波器的姊妹滤波器,称为一阶 FIR 低通滤波器。因此,一阶 FIR 高通滤波器实际上相当于一个低阶 FIR 反梳状滤波器。FIR 低通滤波器则与式 2.3 的正向梳状滤波器相似。此例,使用该音高估计算法所检测到的基频为 551.25Hz,采样率 $f_s = 44.1 \text{ kHz}$ (延迟量 $N = 80$ 个采样点,则基频 $f_0 = f_s / N = 551.25 \text{ Hz}$)。设该长笛音高已经校准并且演奏正确,则该音符的真实基频在 554.365Hz 左右。

需要注意的是,80 个采样点及其倍数是此例平均功率谱曲线的最佳谷距(如图 2.13 的下图)。这一延迟量预测了长笛音符的谐波(或近似谐波)位置,同时也位于与基频呈整数的谷的位置上。此外,从第 40 个采样点开始,延迟量的平均功率谱曲线在第 80 个采样点及其倍数附近都有较大峰值。这一结果是可预料到的,因为反梳状滤波器不仅可以抑制谐波,当陷波频率位于谐波之间时也可放大谐波。因此,如果利用正向梳状滤波器进行基频检测,通过寻找滤波后信号平均功率谱的最大平均能量而不是最低平均能量,也可获得同样的结果,如图 2.14 所示:

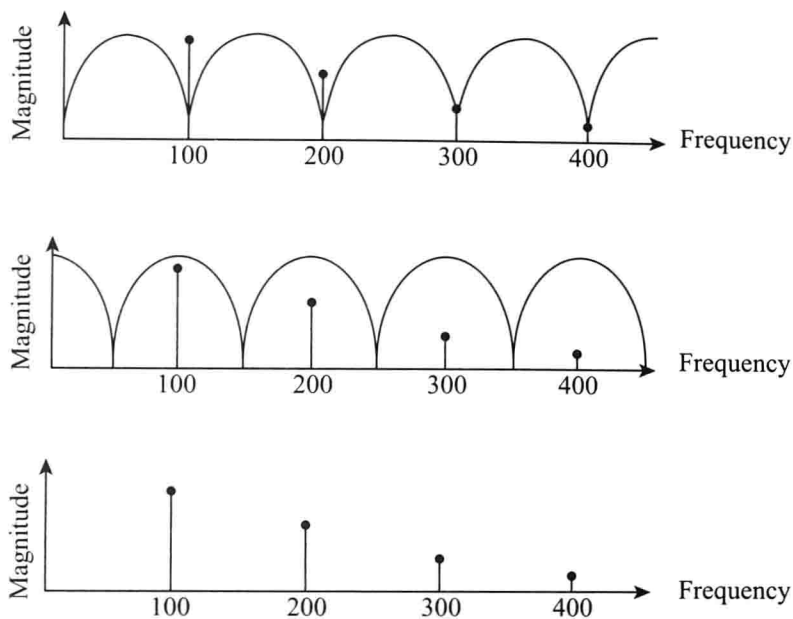


图 2.14 反梳状滤波器

(上图为谐波陷波,中图为经反梳状滤波后的谐波结构,下图为原始谐波结构)

2.4.2 倒谱分析

进行音高分析的倒谱法也基于 DFT,或者更准确地说是基于逆 DFT 变换。该算法的名字本身(Cepstrum)就是由博格特(Bogert)、希利(Healy)和塔奇(Tukey)将频谱(spectrum)的前 4 个字母倒转而得到的,是一个倒谱文字游戏。博格特还以相同的方法创作了其他一些熟悉的术语,如反滤波(liftering)、非谐波(rahmonics)和倒频(quefreny)等。博格特的解释是:

“...we find ourselves operating on the frequency side in ways customary on the time side and vice versa(我们发现在频率方面的研究习惯正好与时域的研究相反,反之亦然)。”(Bogertet al. 1964)

他据此颠倒了字母的顺序。倒谱法最初应用于回波检测中,此后广泛用于声学、语音和其他领域的信号处理。该算法本身非常直观,如式 2.6:

$$\hat{s}[n] = DFT^{-1}\{\log DFT(x[n])\} \tag{2.6}$$

$\hat{s}[n]$ 是具有单位倒频的 IDFT 变换的实数倒谱分量——对应着单位时间,可用于分析峰值以提取基频。

图 2.15 给出了一个短时歌唱声信号的倒谱分析实例。其中,上图显示的是信号的时间序列,中图是以 dB 表示的功率谱,下图为倒谱图。通常情况下,基频就是倒谱中给定区域的最大峰值所对应的频率——即检测限制在音高的最小可能频率和最大可能频率范围内。对于该例的歌唱声的倒谱,最小频率设为 60Hz,最大频率设为 5kHz,最终基频位于 179.27Hz。利用倒谱法检测基频的全过程如图 2.16 所示:

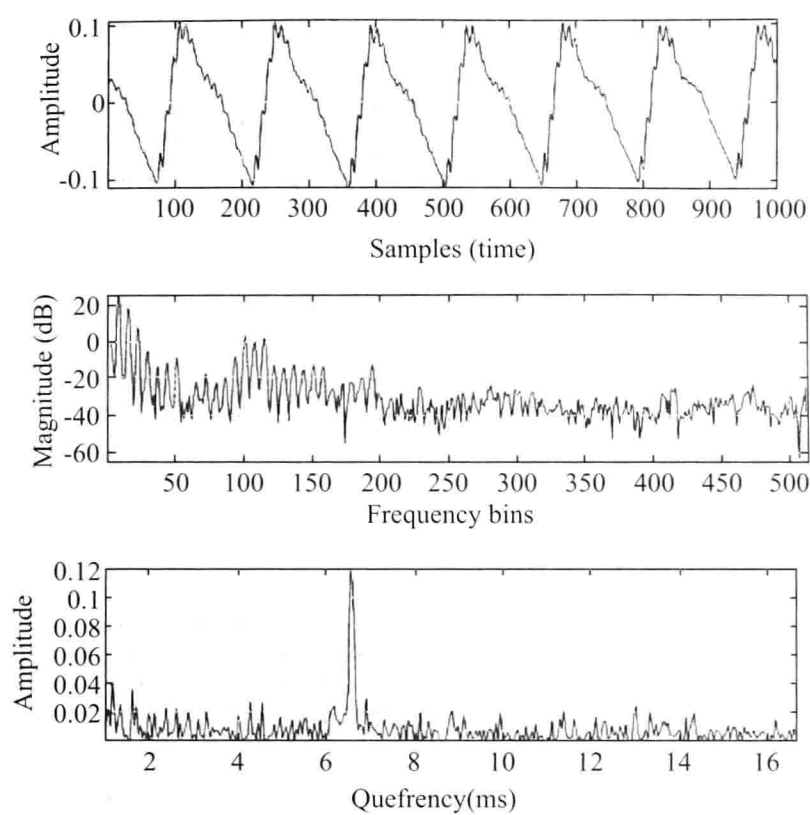


图 2.15 歌唱声,上图为时间序列,中图为功率谱,下图为倒谱

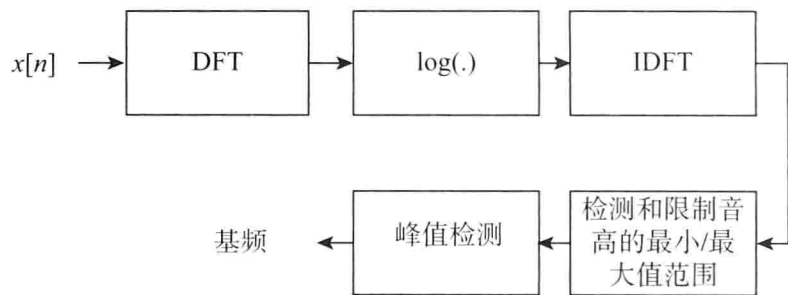


图 2.16 基于倒谱法的基频计算流程图

2.4.3 谐波谱生成法

谐波谱生成法[施罗德(Schroeder), 1968]也是一个基于傅立叶变换的音高检测算法。该算法也是利用有调乐音往往具有一定谐波结构的特点进行的,如图 2.17,该算法通过减采样频谱之间的乘积来检测基频。

频域中的减采样本质上是将谐波转换为 DC 直流分量的过程,如果声音信号的基频及其谐波存在——则 2 倍减采样会将基频的二次谐波(一个八度)转化为 DC 分量,3 倍减采样会将基频的三次谐波转化为 DC 分量,以此类推。然后再将这些减采样谱相乘,则会在基频位置产生一个极大的峰值。对于一个理想的谐波结构系统,所有谐波都位于理想谐波位置(基频的整数倍),且幅值为单位增益 1,而其他位置幅值为 0,则该情况等价于布尔运算系统中的“和”(AND)运算,基频位置是唯一的,且在每个减采样谱中仅出现一次(其他位置幅值为 0)——即基频仅位于一个不会被 0 相乘的点,如图 2.18 所示。不过实际频谱的谐波与基频不可能有完全相同的幅值,此外,其他频率分量的幅值也不可能完全为 0。不过该算法的一般概念对于乐音信号仍然适用,因为乐音的谐

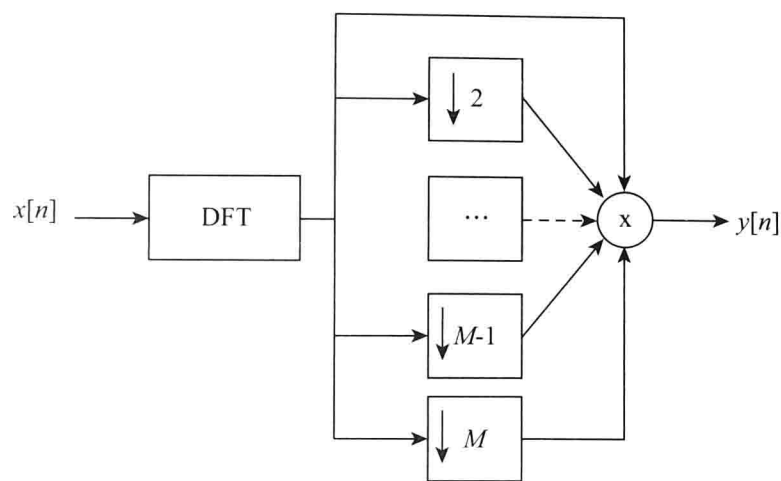


图 2.17 谐波生成算法和减采样

波位置与基频呈近似整数比,同时非谐波分区域的幅值也会明显低于谐波的幅值。因此,一般情况下,找到了上述平移谱所产生的最大峰值就意味着找到了乐音的基频。

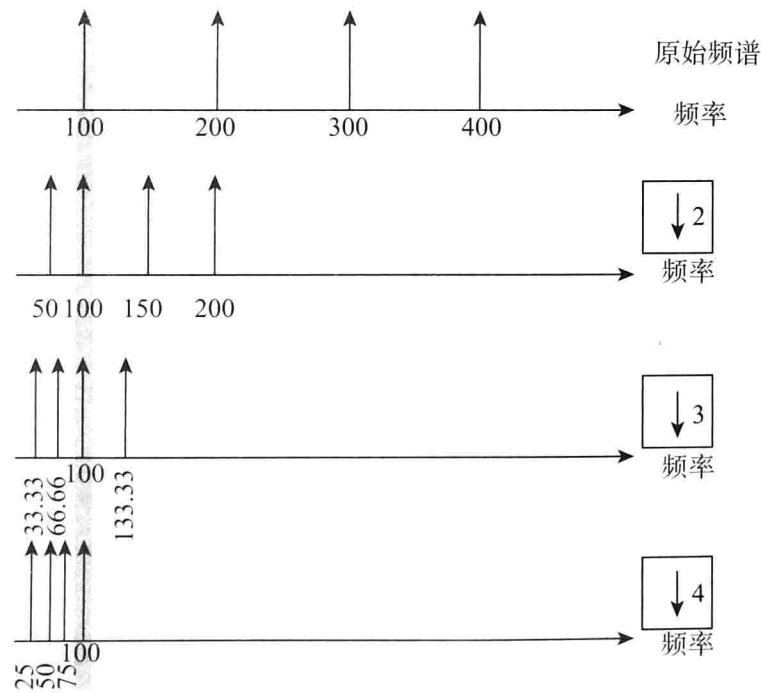


图 2.18 减采样的频谱效应

3. 声 码 器

语音和人声已经是(并且也许将永远是)音乐创作、表演和研究中非常重要的内容。因此,语音和歌唱声的电子合成研究会受到学者的关注也就不足为奇了。声码器(vocoder),即语音编解码器(voice coder-decoder)或语音编解码(voice codec,包括编码和解码器)应用于语音工程领域,最早可追溯到 1929 年间荷马·达德利(Homer Dudley)在美国新泽西州贝尔实验室所进行的工作。一般的声码器由语音编码器和语音解码器组成。编码器部分用于分析语音信号,解码器部分则用于将分析后的信息合成新的语音信号。有关声码器的研究开始于用于通信的人

声合成,最早的演示系统是 1939 年在纽约和旧金山的世界博览会上展出的一个名为 voder (voice operation demonstrator,不要与人名“Darth Vader”混淆)的系统。该系统通过键盘激发一组带通滤波器,以噪声和类脉冲信号合成人声[福朗(Flangan),1972]。这些带通滤波器在本质上模仿了人声的共振峰结构(formant structure)——这些共振频率结构在很大程度上反映了声道的行为和个人声音的音色特征。本节将介绍一些重要的声码器及每个算法背后的基本机制。

3.1 通道声码器

顾名思义,通道声码器(channel vocoder)是一组用于语音分析器及语音合成器的带通滤波器。输入信号(如人声)通过 N 个带通滤波器被分成 N 个频率道通(如图 3.1 所示)这一过程很像家庭和汽车立体声音响系统所使用的均衡器。利用带通滤波的原理在于编码器部分,人声被分成 N 个频率通道或频带,并独立传输至解码器,然后再以来自语音信号的 N 个带通后的通道信息进行加法合成。每个通道本身不会包含太多信息且相互独立(取决于通带的数量)。而当这些通道的信号组合成一个输出信号时,将形成一个与原信号类似的清晰语音信号。如本节绪论部分所述,这一概念由荷马·达德利在 1929 年提出,他最初使用 10 段带通滤波器(通带的中心频率为 250Hz 至 3kHz)来构造一个通道声码器。通道声码器在二战期间也被罗斯福(Roosevelt)和丘吉尔(Churchill)用来规划 D-Day 入侵计划(Dolye 2000),他们借用加密处理的概念,利用该系统进行秘密通话。这种 007 般基于通道声码器的神奇语音系统称为 SIGSALY,它以“绿色大黄蜂”(green hornet)所发出的“嗡嗡”声来合成语音(这也是早期声码器为什么会有“嗡嗡”的背景噪声的原因)。

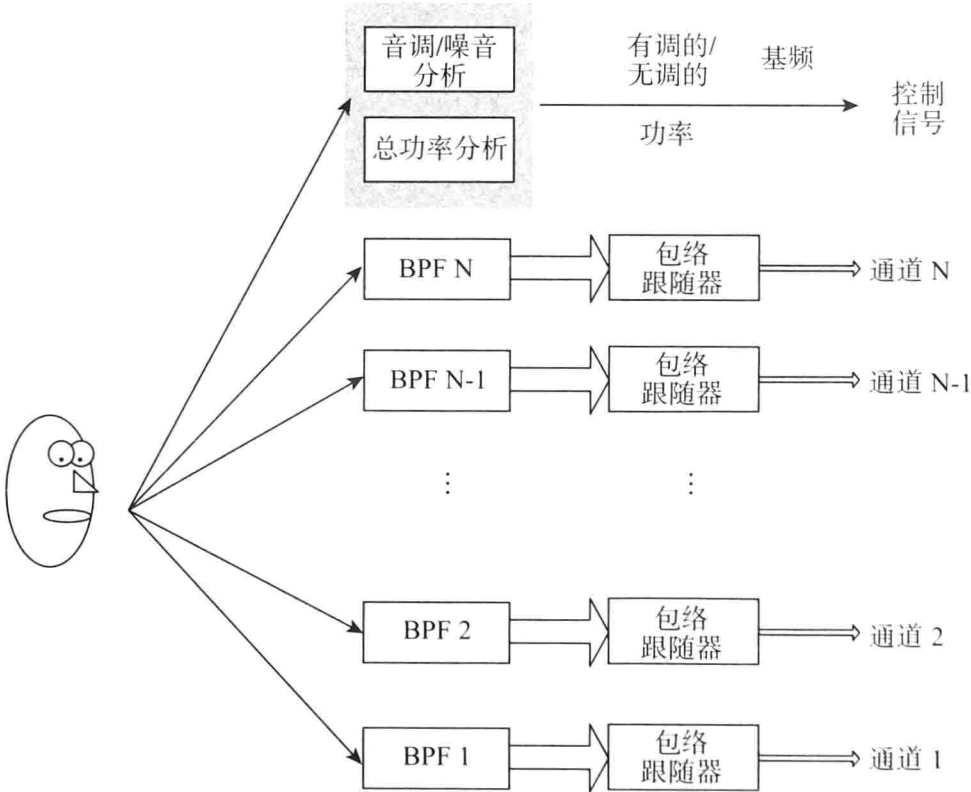


图 3.1 声码器的通道(编码器部分)

3.1.1 滤波带组,包络跟随器及其编码器

回到通道声码器的编码器部分(图 3.1)可知,每个滤波带都有一个包络跟随器。语音信号被划分为 N 个频带后,每个频带都会被输入到一个振幅包络跟随器中进行全波整流(full-wave rectification)和低通滤波处理。全波整流是指在数字系统中对每个样本进行绝对值运算,将负向幅值转换为正向幅值。式 3.1 给出了一个单极低通滤波器的差分方程:

$$y[n] = (1 - a) \cdot x[n] + a \cdot y[n - 1] \quad (3.1)$$

由该差分方程可知,系数 a ($0.0 < a < 1.0$) 是一个权重分配器,输入分量的权重越大,反馈分量的权重越小,反之亦然。

图 3.2 给出了对于不同系数 a 值,式 3.1 包络跟随器所对应的频率响应。包络跟随器有时会取不同的系数,以对信号的某个部分进行激励和衰减。对于信号的激励部分,应当加快包络跟随器的响应以减少低通滤波效果,对于衰减部分,应当产生软着陆效应,使滤波器的输出衰减慢于激励,以获得更强的低通滤波效果。

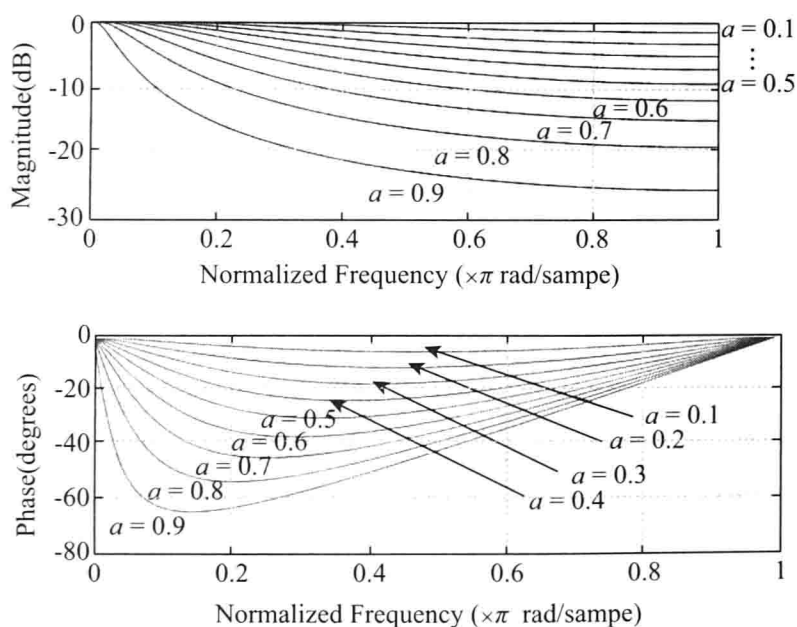


图 3.2 包络跟随器和低通滤波器的频率响应, $a=0.1 \sim 0.9$

因此,对于每个通道,能量级是时间的函数,由包络跟随器表示——而该能量将发送至接收器。信号的总能量级可通过第 2 章第 3.2 节计算振幅包络时使用的 RMS 值计算,如式 3.2:

$$\text{RMS} = \sqrt{\frac{1}{L} \sum_{n=0}^{L-1} x^2[n]} \quad (3.2)$$

通常计算 RMS 值时,需要对信号片断进行时窗分析(长度为式 3.2 中的 L),并像计算 STFT(第 8 章第 5 节)和 RMS 振幅包络(第 2 章第 4 节)那样设置合适的跳距。图 3.3 给出了通过 MATLAB® 巴特沃斯带通滤波器计算的 24 个滤波带的频率响应,其中频段的中心频率与人耳频率感知的 Bark 尺度相一致(详见附录)。同时,也可对滤波带使用线性尺度。不过由于人耳的听觉系统是非线性的(对频率与幅值的感知都是非线性的),对于可听声信号处理, Bark 等的非线性尺度更加适用(Zwicker, Hastl 1999)。这些中心频率位置对应的频段称为临界带(critical

band), Bark 尺度正是符合人耳频率感知倾向的尺度之一。临界带表示了给定中心频率对应的带宽(由上下限临界频率定义),并激励基底膜的对应区域(位于内耳耳蜗内的纤维组织上),其响应的输入正弦波频率位于该中心频率上。例如对于 45Hz 正弦波,由 Bark 尺度可知,对其他介于 49Hz 和 51Hz 之间的强度相同的正弦波的感知,并不能与 45Hz 的正弦信号进行区分。临界带宽随中心频率而变化,并随该中心频率的增大而变宽。

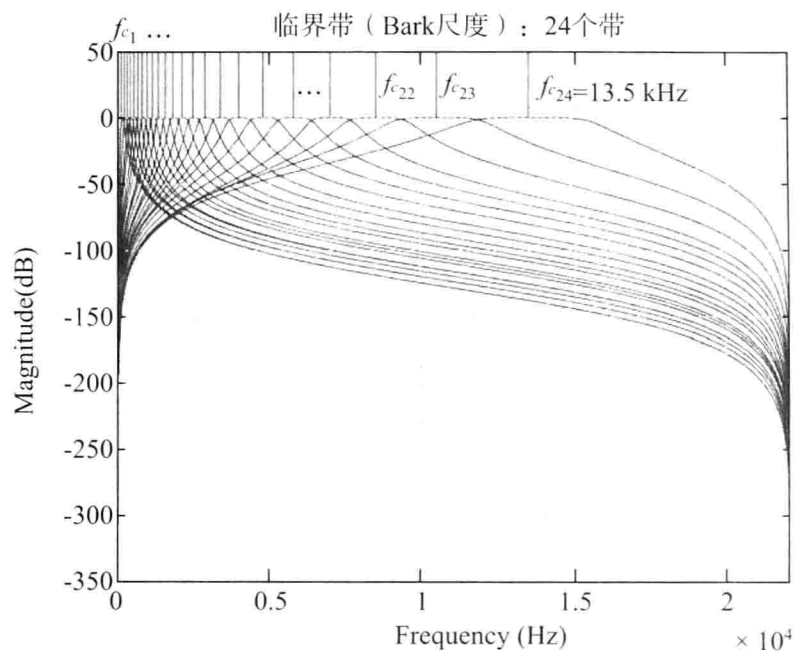


图 3.3 24 个 Bark 滤波器

由图 3.4 可知包络跟随器的作用效果。其中,上图为输入人声的波形,后面的图形是经过一个 24 带包络跟随器后的输出信号。由该图可知,每个滤波带在不同时刻的能量不同。仅仅当所有频带在编码器部分重新叠加后,才能得到一个完整的、非间断的合成语音信号。

3.1.2 语音与非语音成分分析与解码

在开始介绍通道声码器的解码器之前,我们还没有讨论编码器中有调声与噪声成分的分析过程。人们在彼此对话时也许会惊奇地发现,自己发出的说话声是有调声与无调声的混合声。辅音或非语音成分组成了类噪信号,而说话声中的元音则显示出强烈的周期性和有调性,或称为语音成分。尝试在任何发音舒服的音域演唱字母“s”或“f”,然后再尝试在该音域演唱字母“a”或“e”,此时会发现对于前两个字母很难保持一定的音高,而对于后两个字母,演唱则确实是有调的。这就是人们可能没有注意到的语音信号的基本特征,同时也在不知不觉中使语言成为一种潜在的、富于音乐性的信号。

回到通道声码器,在了解了编码器中与有调声和噪声分析模块有关的问题后,到目前为止,为什么需要判断语言信号是语音的还是非语音的呢?原因是通道声码器与一个特定的合成模型有关,它称为信源滤波器(source-filter)模型。该例的源信号本身就包含了语音与非语音成分,相应的带通滤波器也对应着信源滤波器模型结构的滤波器部分。因此,在这样一个系统中,语音合成阶段是由源信号激发解码端的滤波带器组而完成的[有时甚至会在编码端完成这一过程——称为合成分析(analysis-by-synthesis)]。根据不同的信源类型(语音或非语音成分),将由一个开关决定产生一个脉冲序列还是噪声信号。从某种程度上说,解码器利用了人类语言

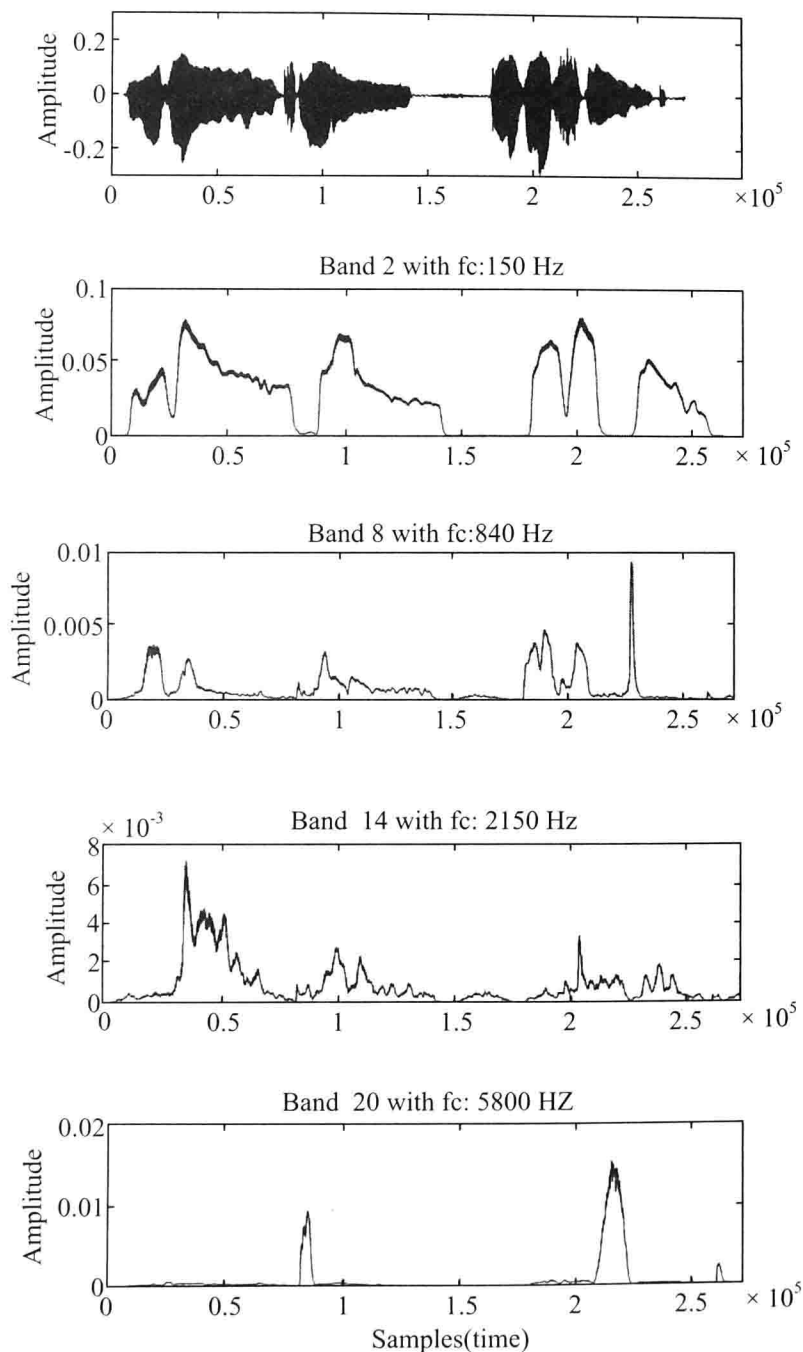


图 3.4 四个频带的跟随包络

(最上图完整波形,第二图 $f_c = 150\text{Hz}$,第三图 $f_c = 840\text{Hz}$,第四图 $f_c = 2.15\text{ kHz}$,第五图 $f_c = 5.8\text{ kHz}$)

系统的信源滤波特征来解决语音合成问题——即沿着振幅包络,通过语音和非语音成分的判定标志提取和接收来自于发射器或者编码器的基频信息,以驱动滤波带中的各个滤波器。被分析的信号为非语音成分时,解码器中各滤波带的输出噪声信号,为语音成分时,带通滤波器组则生成一个脉冲序列。然后将每个滤波器的输出叠加,即可得到合成语音信号。这一思想总结如图 3.5 所示。

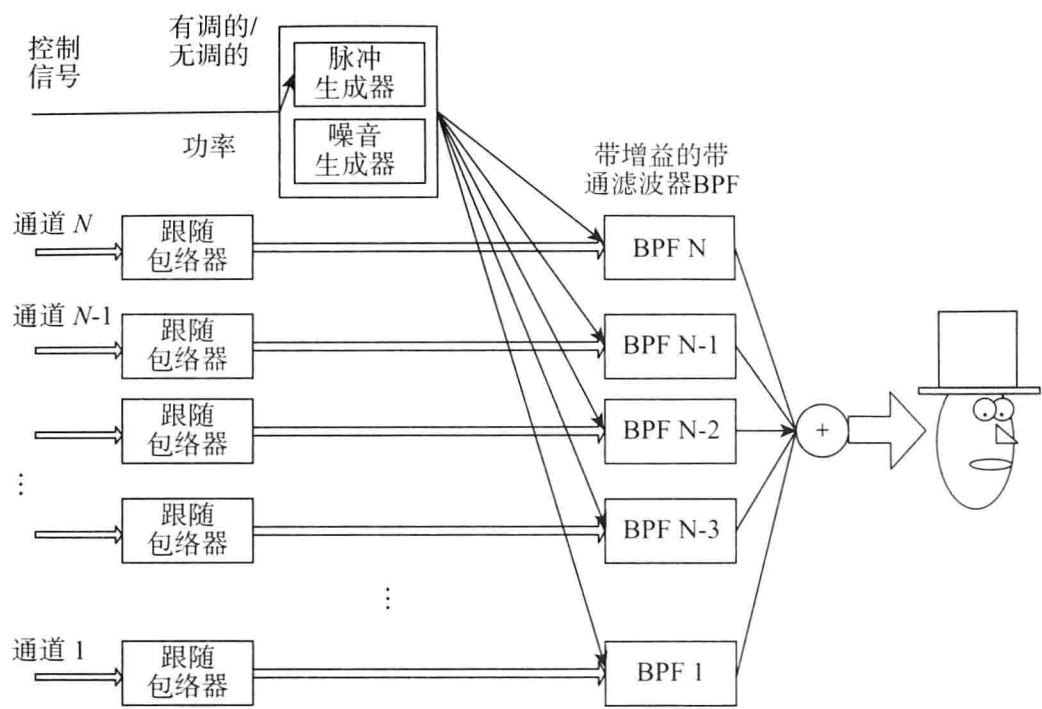


图 3.5 声码器通道(解码器部分)

同时也需要注意的是,图 3.1 和图 3.5 都没有将实际的音频信号传输到解码器。发送至解码器的信号仅由每个滤波带的能量包络特征、判断语音成分和非语音成分的控制信号,以及给定时窗片断的总能量组成。声码器的重点在于编码端的原因是,它需要通过有限的带宽传输用于合成的原始信号,目标是在保证需要的情况下降低传输的信息量。声码器主要用于语音通信和移动电话技术领域,在很大程度上依靠语音数据压缩技术来传输必要信息,以减少数据的发送量。

3.1.3 语音与非语音成分判定

退一步说,目前还没有讨论实际上如何分析信号的语音和非语音成分。因此,本节将从最简单的算法——过零率开始介绍一些常见的语音与非语音成分分析算法。

3.1.3.1 过零率分析

早些时候,在第 2 章第 5.1 节介绍了用于基频检测的过零率分析方法。该算法同样也可用于判断信号的语音与非语音成分。过零率本身的概念非常明确,如果将第 2 章所学到的方法与声信号的基本知识结合起来,过零率算法对该问题的解决也非常有效。众所周知,理想类噪声信号(非语音成分)是无规律的或非周期性的。换句话说,周期信号(语音成分)因具有重复周期模式而呈现出循环趋势,如图 3.6。所以,非语音成分的过零率“非常高”,而语音成分的过零率“很低”。称一些事物低和高有时候相当于说它们“暖”和“冷”,不过这些定性判断没有太大意义,除非进一步做定量分析。但即使定量分析结果出现在我们的计算机屏幕上,也不可能肯定地判断一个信号是语音的还是非语音的。不过,过零率的范围往往表示了语音或非语音特征,如图 3.7 所示:

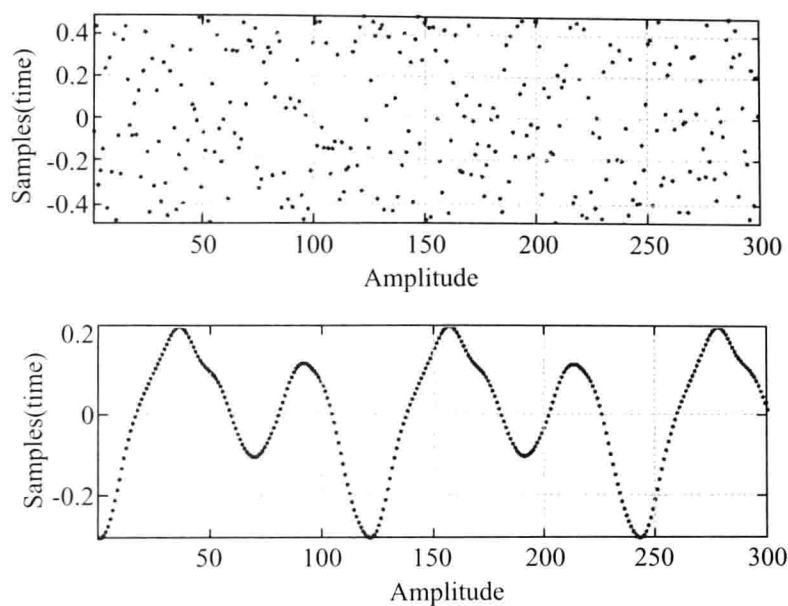


图 3.6 噪声(上图)与周期信号(下图)

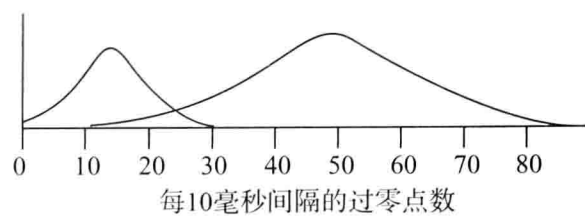


图 3.7 语音和非语音信号的 ZCR 分布

图 3.7 给出了用宽度为 10 毫秒的时窗绘制的语音与非语音信号的平均过零率 ZCR 分布图 (Rabiner 和 Schafer 1978)。虽然两者的图像有些重叠,但非语音成分的平均过零率高斯分布中心位于每 10 毫秒约 49 个,而语音成分则为每 10 毫秒约 14 个。图 3.8 给出了歌唱声“Love me

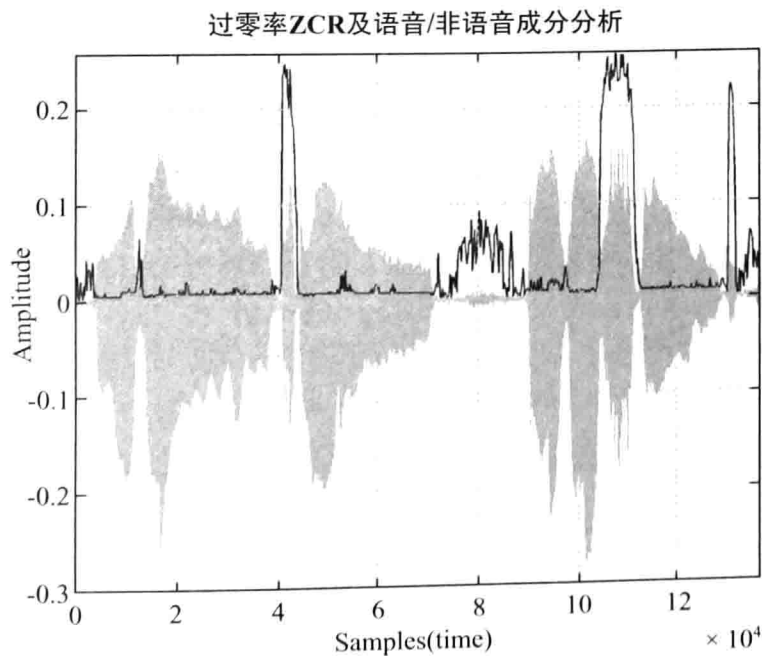


图 3.8 歌唱声的波形与过零率图像叠加结果

tender, love me sweet”的波形和用上述时窗计算的过零率时域分布图。由图不难看出,此例的噪声成分约在以下位置:位于“tender”的“t”音上(第 40,000 个采样点附近),短暂停顿之后(第 72,000 个采样点附近),位于“sweet”的“s”音上(在第 104,000 个采样点附近)。

图 3.9 给出了整个波形时域的实际过零率分布。图 3.10 则显示了约从第 40,000 个采样点开始的“tender”中“t”音的过零率。由图可清晰地区分出两个反应语音和非语音成分波形特征的区域——第一部分是无特定周期的噪声部分,第二部分是过零率低、波形呈模式化准周期特征的语音成分。

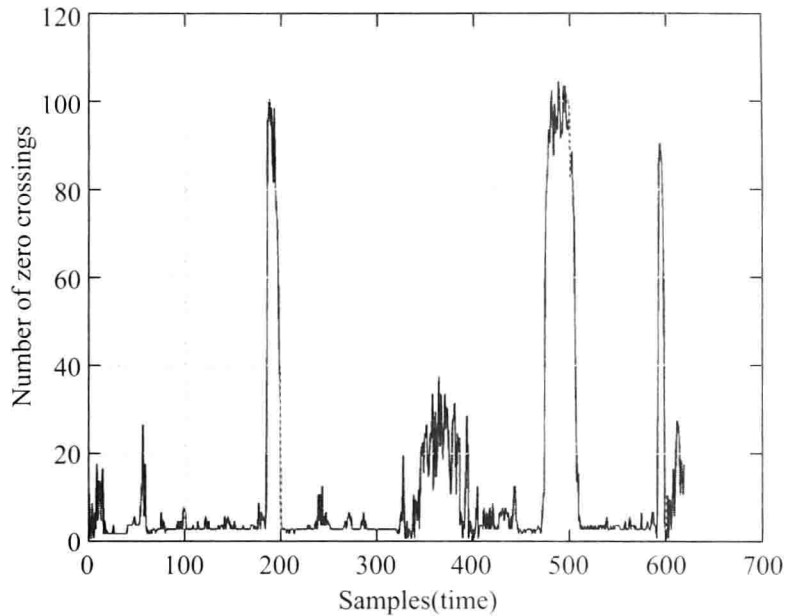


图 3.9 10 毫秒时窗的过零率分布

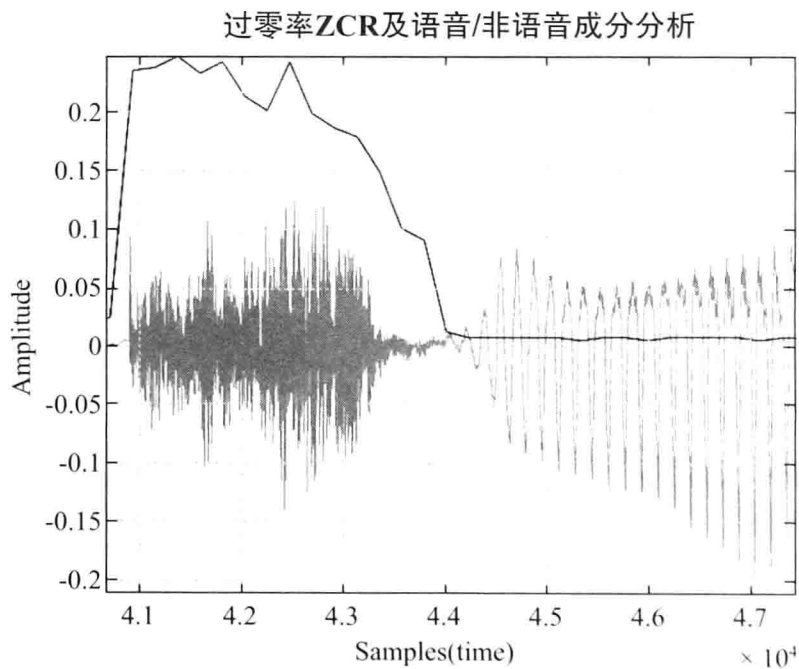


图 3.10 放大后的“ten(der)”波形

3.1.3.2 预权重能量率

用预权重能量率(pre-emphasis energy ratio,Knodoz 2004)对语音与非语音成分进行分析的

重点是关注相邻两个采样点的幅值差的变化。相较于信号的非语音区域,语音成分该差值的变化相对较低——与图 3.6 相一致,它显示了语音信号各采样点的幅值变化缓慢而平滑,典型非语音信号的幅值则呈随机分布。该算法的表达式如式 3.3,其中分子部分采用了一个预权重高通滤波器,因此该算法被称为预权重能量率(见式 2.5)。

$$\text{Pr} = \frac{\sum_{n=1}^{n=N} |x[n] - x[n-1]|}{\sum_{n=1}^{n=N} |x[n]|} \quad (3.3)$$

图 3.11 给出了该算法对上例歌唱声样本进行判定的结果,时窗长度为 10 毫秒。

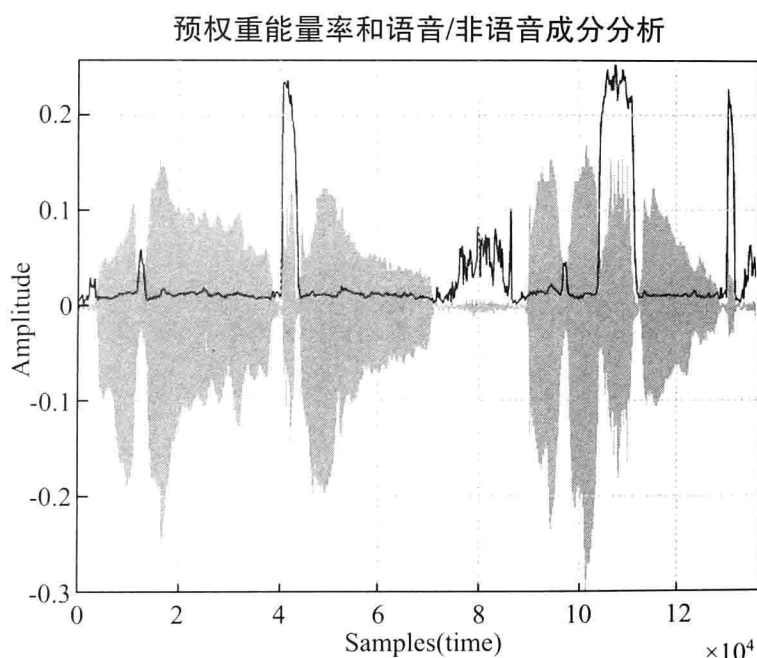


图 3.11 预权重能量率法对波形的分析结果

3.1.3.3 低频能量率

在语言声中,语音成分的能量往往会集中于低频,而非语音成分则集中在高频。不仅语音信号有这一特点,对于乐音信号,特别如本章和前几章所述的声学乐器的乐音信号也是如此——即谐波能量会随着频率增加至奈奎斯特限制频率而衰减。则低频能量率 LF (low-band to full-band energy ratio) 算法通过分子的低通滤波器来计算低频参量与原信号全频能量的比值,如式 3.4。当输入信号 x 为语音成分时, x 和 x_{LPF} 的能量或多或少会相同,则式 3.4 的比值将接近 1。这是因为对于语音信号,式 3.4 分子部分的低通滤波器几乎没有作用。而如果输入信号 x 为非语音成分,输入信号 x 将表现出平坦的频谱(理想噪声信号的频谱是平直的), x_{LPF} 将位于较低的能量级。总之,对于语言信号的非语音区域,低频能量率 LF 低较,对于语音区域,低频能量率 LF 则较高。

$$LF = \frac{\sum_{n=1}^{n=N} x_{LPF}^2[n]}{\sum_{n=1}^{n=N} x^2[n]} \quad (3.4)$$

图 3.12 给出了采用低频能量率算法对上例歌唱声信号进行判定的结果,时窗长度和之前

一样为 10 毫秒。注意到在此例中,非语音成分具有较低能量,语音成分具有较高能量。

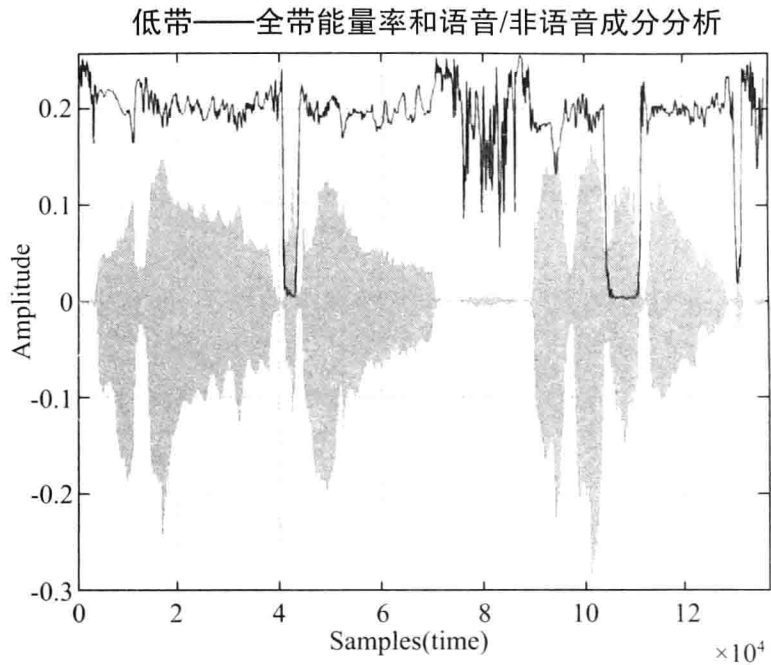


图 3.12 低频全频能量率法对波形的分析结果

3.1.3.4 频谱平坦量检测

频谱平坦量检测 SFM(spectral flatness measure) 也可用于判断语音与非语音成分,该参数通过信号 DFT 变换的几何平均值 G_m 与算术平均值 A_m 的比来计算,如式 3.5。如果信号具有类似正弦或余弦的周期性,则 SFM 值较低(接近于 0),而如果信号具有类噪特征(频谱较为平直且自相关性很低),则 SFM 值接近于 1。

$$SFM_{dB} = \frac{Gm}{Am} = \frac{\left[\prod_{k=0}^{N-1} X^d(k)\right]^{\frac{1}{N}}}{\frac{1}{N} \sum_{k=0}^{N-1} X^d(k)} \tag{3.5}$$

$$\prod_{k=0}^{N-1} X^d(k) = X^d(0) \cdot X^d(1) \cdot \dots \cdot X^d(N-2) \cdot X^d(N-1) \tag{3.6}$$

需要注意的是,该算法在计算几何平均值时是先求乘积再取 $1/N$ 次幂,而计算算术平均值时则是先求和再进行除法。由于在计算几何平均值时需要乘法,因此在计算之前应该剔除零值,无论其他项的值有多大,都要避免零值项使几何平均值恒为零的情况。同时,由式 3.7 可知,算术平均值的结果总是大于或等于几何平均值,其中系数 a_N 为正数。对于非语音成分,上述几何平均值和算术平均值近似相等,SFM 值接近于 1,对于语音成分,几何平均值会变小从而降低了 SFM 值。

$$\frac{(a_1 + a_2 + \dots + a_{N-1} + a_N)}{N} \geq (a_1 \cdot a_2 \cdot \dots \cdot a_{N-1} \cdot a_N)^{\frac{1}{N}} \tag{3.7}$$

利用对数恒等式(对数的和等于各真数乘积的对数,真数幂的对数等于真数的指数与真数对数的乘积),几何平均值 G_m 可表示为式 3.8。该式有时又称为对数平均值(log-average),因为

在进行指数运算之前提取了对数以还原各项的原始尺度。

$$\left[\prod_{k=0}^{k=N-1} X^d(k) \right]^{1/N} = \exp \left(\frac{1}{N} \sum_{k=0}^{k=N-1} \ln X^d[k] \right) \tag{3.8}$$

图 3.13 给出了 SFM 法对上述歌唱声信号进行判定的结果,时窗宽度仍为 10 毫秒,采用非重叠的 STFT 频率帧进行计算。由图可知,该算法的判定结果更为精细,更小的时窗会降低频率分辨率。而增大时窗(提高频率分辨率)并保持跳距不变,SFM 曲线将变得更细致,如图 3.14 所示。该图所使用的参数为时窗宽度为 11.61 毫秒(512 个采样点),STFT 变换的重叠率为 10 毫秒(220 个采样点),采样频率 $f_s = 44.1 \text{ kHz}$ 。

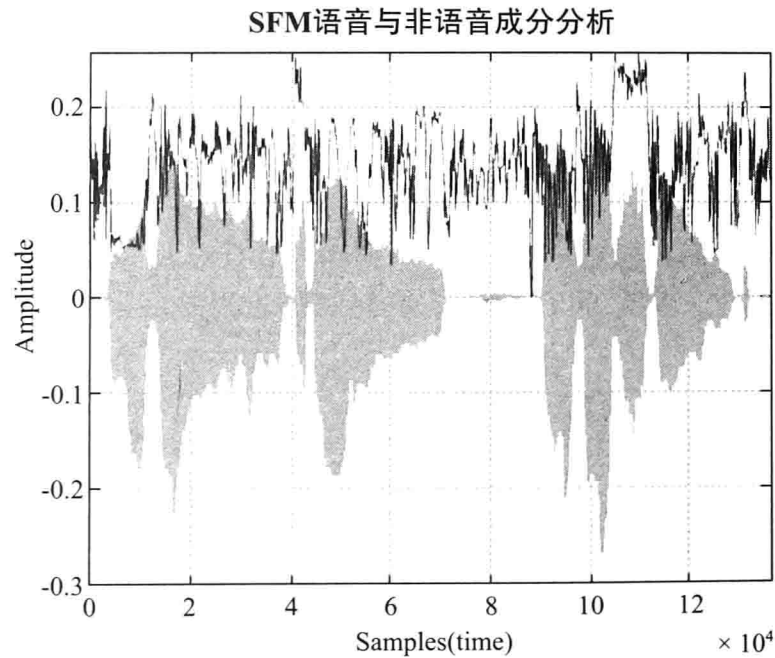


图 3.13 语音成分判定的 SFM 分析,时窗宽度为 10 毫秒

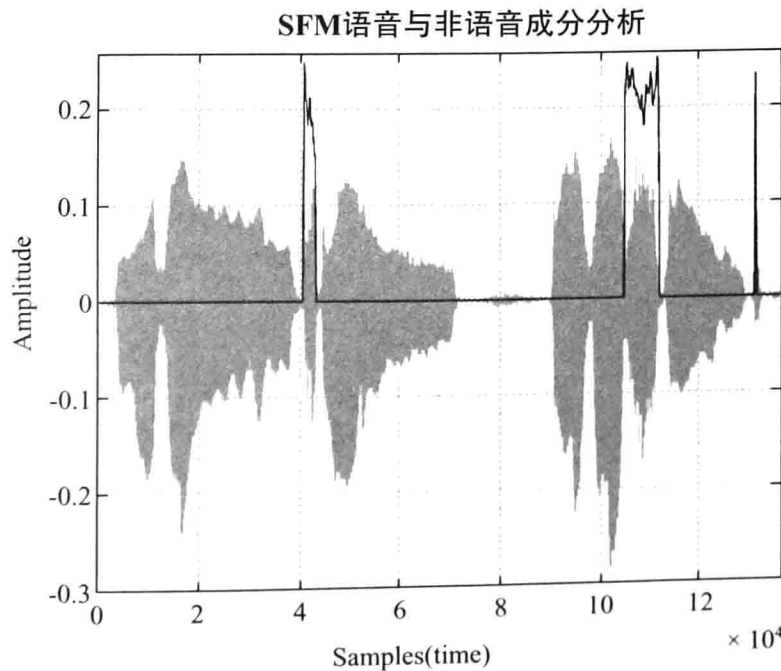


图 3.14 语音成分判定的 SFM 分析,时窗宽度为 40 毫秒

3.2 线性预测编码

本节接下来要介绍的是线性预测编码 LPC (linear predictive coding) 声码器,它是在 20 世纪 60 年代中后期,由比什努·S.阿塔尔 (Bishnu S. Atal) 在贝尔实验室跟随马克斯·马修斯 (Max Mathews, 第一个计算机音乐语言的发明者) 工作时率先提出的。线性预测编码在早期被称为自适应预测编码 (adaptive predictive coding), 后沿用 LPC 一词。与通道声码器一样, LPC 源于对语音的研究, 后来流行于计算机音乐领域, 尤其见于作曲家查尔斯·道奇 (Charles Dodge) 和保罗·兰斯基 (Paul Lansky) 的早期计算机音乐作品中。该理论本身可被看成一个分析/合成算法, 由式 3.9 可知, 适当设置 FIR 滤波器的权重时, 它以过去的 P 个样本来预测当前样本 $x[n]$ ——即当前样本是过去输入采样的线性组合。

$$\begin{aligned}\hat{x}[n] &= a_1 \cdot x[n-1] + a_2 \cdot x[n-2] + \cdots + a_P \cdot x[n-P] \\ &= \sum_{k=0}^{k=P} a_k \cdot x[n-k]\end{aligned}\quad (3.9)$$

$\hat{x}[n]$ 为当前估计样本, $x[n-k]$ 表示延迟量为 k 的过去样本, a_k 为过去样本的权重系数。由上式可知, 线性预测编码的目标是寻找合适的 P 值和权重系数 a_k , 使当前估计样本 $\hat{x}[n]$ 尽可能地接近实际样本 $x[n]$ 。因此, 为了减少 $\hat{x}[n]$ 与 $x[n]$ 的误差 $e[n]$, 称为 LPC 系数的最优权重的计算如下:

$$e[n] = x[n] - \hat{x}[n] = x[n] - \sum_{k=1}^{k=P} a_k \cdot x[n-k] \quad (3.10)$$

从某种程度上说, 由过去事件预测当前事件是一种常规思路 (虽然人们往往不会从过去的错误中学到更多东西。) 由于 LPC 算法通常用来预测短期趋势 (20~30 毫秒), 不适用于预测长时特征 [称为长时预测 (long-term prediction), 如音高估计], 因此也被称为短时预测滤波器 (short-term prediction filter)。

由式 3.9 可知, 该滤波器本身是一个 FIR 系统, 其误差 $e[n]$ [通常称为残差 (residual) 信号] 的 z 变换如式 3.11, 其传递函数为 $A(z)$, 如式 3.12:

$$\begin{aligned}E(z) &= X(z) - \hat{X}(z) \\ &= X(z) - X(z) \cdot \sum_{k=1}^{k=P} a_k \cdot z^{-k}\end{aligned}\quad (3.11)$$

$$\begin{aligned}&= X(z) \cdot \left(1 - \sum_{k=1}^{k=P} a_k \cdot z^{-k}\right) \\ A(z) &= 1 - \sum_{k=1}^{k=P} a_k \cdot z^{-k}\end{aligned}\quad (3.12)$$

因此, 输入信号的 z 变换 $X(z)$ 可表示为传递函数 $A(z)$ 和残差信号 $e[n]$ 的 z 变换 $E(z)$ 的函数, 如式 3.13:

$$X(z) = \frac{E(z)}{A(z)} \quad (3.13)$$

LPC 算法的分析模块与合成模块结合使用时功能是很强大的, 这一系统也是一种分析—合成结构。其分析模块是一个全零点 FIR 滤波器, 提取 FIR 滤波器系数并求倒数, 则 FIR 滤波器

将变成一个全极点 IIR 滤波器,如式 3.14:

$$\frac{1}{A(z)} = \frac{1}{1 - \sum_{k=1}^{k=P} a_k \cdot z^{-k}}$$

(3.14)

全极滤波器实质上是一个描述声道的共振滤波器。因此,由式 3.13 可知,如果已知误差 $E(z)$ 和滤波器 $A(z)$ 可计算 $X(z)$,则在传递函数 $1/A(z)$ 中加入误差信号可重构输入信号 $x[n]$ 。也就是说,滤波器 $1/A(z)$ 受残差信号 $e[n]$ 激励时(构成通道声码器的信源滤波模型),可得到输入信号 $x[n]$ 。

声道相当于一小段传播人声的管道,LPC 模型非常适合描述这些声道以及人类语言声的共振峰(formant)结构。共振峰指共振频率点,典型范围是人声基频的前 3 至 5 次谐波,其位置沿频谱频率轴分布(有点像一顶帐篷被杆子撑成的特定的形状——即帐篷共振峰所组成的形状)。由于声管模型是 LPC 系统的物理基础,某些乐器会表现出比其他声源更强的共振特性。如双簧管或单簧管等木管乐器,由于或多或少是管状乐器,一般能显示出一个或两个明显共振区域。

LPC 系统的信源滤波器模型如图 3.15 和图 3.16。解码器是滤波器模型中的激励信号发生

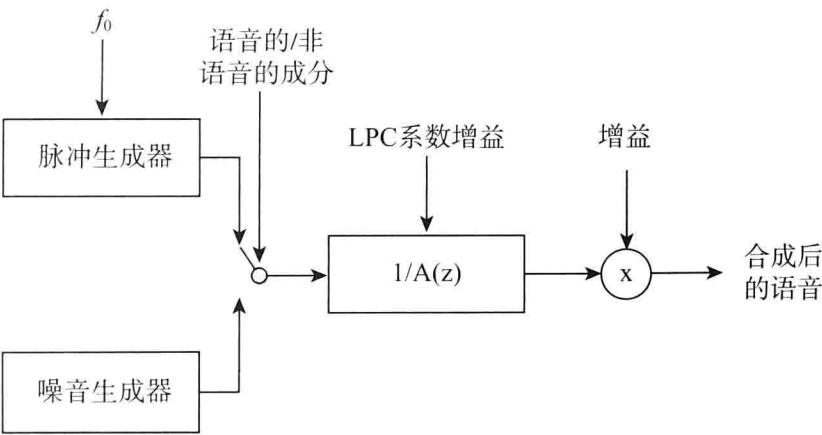


图 3.15 LPC 系统的合成模块

器(生成语音或非语音成分),编码器用于信号分析,并将所得的相关信息传送至解码器进行合成。由图可知,该系统的总体结构与通道声码器相同。提供给全极滤波器的激励信号发生器可以是一个脉冲发生器或噪声发生器,以生成语音或非语音信号。如上节所述,需要在脉冲声和噪声之间进行切换的原因是,信号为语音成分时(如元音)有调,为非语音成分时,激励信号(残差信号 $e[n]$)为平直随机噪声。

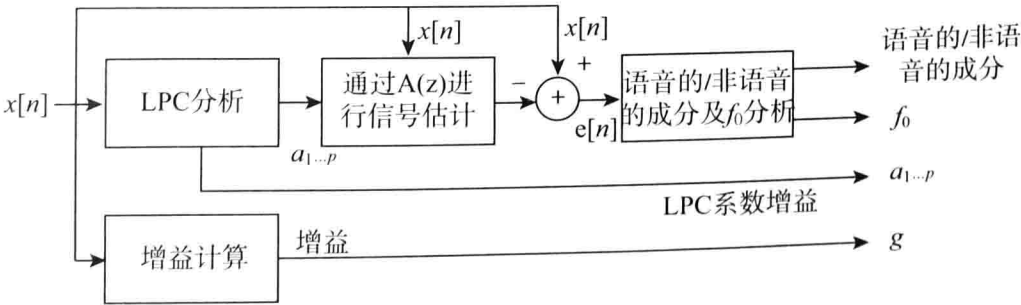


图 3.16 LPC 系统的分析模块

因此,如果分析模块可用自相关法分析残差信号 $e[n]$ 而得到基频信息,则在合成模块可用该基频脉冲序列来激发全极滤波器。如果是无调声,可使用噪声信号来激发该滤波器。语音和非语音成分判定算法详见 3.1.3 节。对于时窗分析所截取的信号片断,增益 g 用 RMS 等方法计算。对于给定时域,该增益为选定值,以使输出(合成信号)和输入(原始信号)的增益相同。这种脉冲声/噪声激励滤波器系统多用于低比特率声码器(编码器与解码器),常见于移动电话系统,如 GSM 制式手机。尽管该声码器的总体结构并无与众不同,但其中一个特别之处是,LPC 编码器的输入信号本身是合成信号并用于编码过程。因此,该编码器实际上既是一个分析器也是一个合成器,是一个真正的 AbS 分析合成(analysis-by-synthesis)系统。而图 3.16 并不是一般 LPC 声码器的完整结构,因为在现实应用中,只有残差信号会发送至编码器并用于合成。如果为了节省带宽,在合成过程中不发送残差信号的声码器称为码表声码器(codebook vocoder)——如码表激励线性预测 CELP (codebook excited linear predication) 声码器就是一个很好的例子。因为从数据压缩的角度看,残差信号的发送(即使只是原信号的一部分)非常浪费传输带宽,所以 CELP 会在编码端分析残差信号然后向码表传送一个索引信号,而不是真正的残差信号。码表在编码和解码端均已预先设定,它包含了一系列在解码端的重新合成阶段用于激励全极滤波器的最佳拟合残差信号。因此,相对于真正的残差信号,索引信号的发送具有很高的数据压缩量。图 3.16 还忽略的另一个模块是编码器分析前端的预权重滤波器(pre-emphasis)和解码器重构后端的去权重滤波器(de-emphasis filter)。本章 2.4.1 节已将预权重滤波器做为最简的 FIR 高通滤波器(式 2.5)进行了介绍。在 LPC 分析之前加入预权重滤波器的原因,是考虑到频谱能量在频率轴上的分布均衡时(语言声和乐音的低频能量密度较高),LPC 系数将具有更好的鲁棒性。通过高通滤波,预权重滤波器有助于频谱能量的均匀分布。而为了还原信号的频谱,则需要在重构信号之前加入去权重过滤器。去权重滤波器是一个最简低通滤波器,如式 3.15:

$$y[n] = x[n] - b_1 \cdot x[n - N]$$

(3.15)

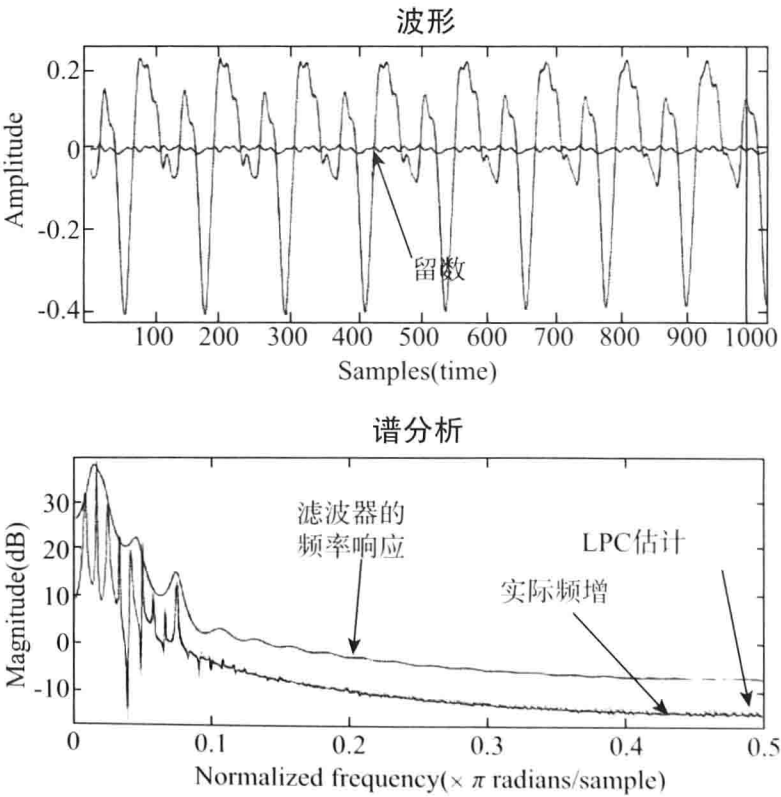


图 3.17 电贝斯的 LPC 分析实例

图 3.17 给出了一个电贝斯信号的 LPC 分析与重构结果。由图可知,估计信号与原始信号的时域波形和频谱几乎相同。图 3.17 的上图给出的残差(误差)信号进一步说明了这一点。图 1.7 的下图还给出了所使用的全极滤波器的频率响应,以及原信号和估计信号的 DFT 变换,两条曲线非常接近,几乎没有什么区别。综上所述,如果将 LPC 做为一个声码器,用于编码/解码系统、语音与非语音成分判定模块、频率检测模块以及解码器中全极滤波器的激励信号发生器,那么它在音乐信号处理中的应用前景将十分乐观。因为有了为上述小项目所做的基础知识储备,对 LPC 的深入讨论可留做周日晚上实验室的作业(我们不希望这些奇怪的声音打扰了隔壁不知情的邻居)。

3.3 LPC 系数的计算

由于很多语音信号处理和数字技术著作对 LPC 系数的算法已有详细介绍,因此本节不再深入讨论,只简要介绍两种最流行的 LPC 系数(式 3.5 中的 $a_1 \cdots a_p$)算法——自相关法(autocorrelation)和协方差法(covariance method)。

自相关法是一种基于语音帧的自相关向量来计算 LPC 系数的算法。自相关向量由莱文森-杜宾算法(Levinson-Durbin algorithm)提取,通过递归和有效自相关逆矩阵向量求解 LPC 系数。该算法需要时窗具有锥形边界——如侧翼起始与结束端点幅值均为 0 的汉恩窗。在处理过程中,该算法虽然会损失一些信息,并使信号在频域产生失真,但它因保证了滤波器的稳定性而使 LPC 系数的计算更为准确。需要记住的是,重构滤波器是一个全极滤波器,为了保证系统稳定,系统的根或极点需位于单位圆内——这样的滤波器称为最小相位滤波器(minimum-phase filter),它的所有根都位于单位圆内。该算法如下式 3.16 和 3.17,其中 $r[\cdot]$ 是自相关向量, a_k 是 LPC 系数。

$$r[\tau] = \sum_{n=0}^{n=N-1} x[n] \cdot x[n+\tau], \tau \geq 0 \quad (3.16)$$

$$\sum_{k=1}^{k=P} a_k \cdot r[|k-m|] = -r[m], \quad m = 1, 2, \dots, P-1, P \quad (3.17)$$

与自相关法不同,协方差法计算 LPC 系数时不需要进行时窗分析,该算法由决定人声音高的周期始发端采用基音同步法进行分析。相较于自相关法,采用协方差法计算 LPC 系数具有更高的更新率,更能适应声道传递函数的变化,计算结果也更精确。不过,协方差法需要更多的计算量,如式 3.18 和 3.19,其中 c_{km} 为协方差矩阵,通常用 Choleski 分解法来求解系数 a_k (Rabiner, Schafer 1978)。

$$c_{km} = \sum_{n=P}^{n=N-1} x[n-k] \cdot x[n-m] \quad (3.18)$$

$$\sum_{k=1}^{k=P} a_k \cdot c_{km} = -c_{0m} \quad m = 1, 2, \dots, P-1, P \quad (3.19)$$

对于音乐信号,基于信源滤波模型结构,可以按需要改变基频或激励信号,甚至是任何原信号的残差信号。因此,从音乐家的角度看,对于音乐创作与分析,不仅局限于语音信号也指其他一般音频信号,LPC 系统也是一个非常强大而灵活的工具。信源滤波模型本质上是与第 7 章第 4 节提到的“语音共振箱”模型是一致的。在“语音共振箱”模型中,电吉他声就是一个信源,该

信号进入模拟人声道共振结构的盒子腔体后经过滤波产生吉他说话的效果。使用相应的 LPC 系数可以在数字系统实现这种“语音共振箱”效果——这是一种潜在的音乐趣味。

3.4 相位声码器

简而言之,相位声码器(phase vocoder)是最强大的声码器之一,就算它不是最好的声码器,也堪称早期同类通道声码器更年轻、更先进且更灵活的声码器。对于该类声码器的研究,较早见于弗拉纳根(Flanagan)和高登(Golden)在贝尔实验室的开创性工作(Flanaga 和 Golden 1966,正如大家所知,那些年在贝尔实验室中有许多不可思议的研究者)。虽然最初的研究应用目标是语音,但随着技术的发展,相位声码器在电子及计算机音乐领域中越来越流行。

相位声码器与通道声码器相似,也是基于滤波带的思想,但其带宽规模更大。该合成器本身在解码端没有使用激励信号模型(语音与非语音激励信号),而在解码器的窄带校准滤波带中使用瞬时幅值信息来构造加法合成系统——即用特定的振幅、频率和相位参数来合成波形。对于通道声码器,滤波带振幅是时间的函数(瞬态幅值),同时并不关注自身的频率变化——频率与相位信息在两个声码器中分开设。相位声码器则包含了由两个相邻时间帧的瞬时相位(instantaneous phase)所计算的瞬时频率(instantaneous frequency)。这对于合成“高品质”的信号是至关重要的,如图 3.18、图 3.19 和图 3.20:

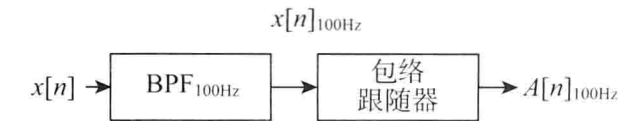


图 3.18 单通道声码器,分析模块,中心频率为 100Hz

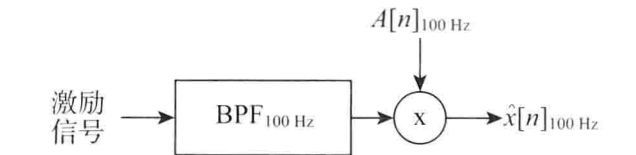


图 3.19 单通道声码器,合成模块,中心频率为 100Hz

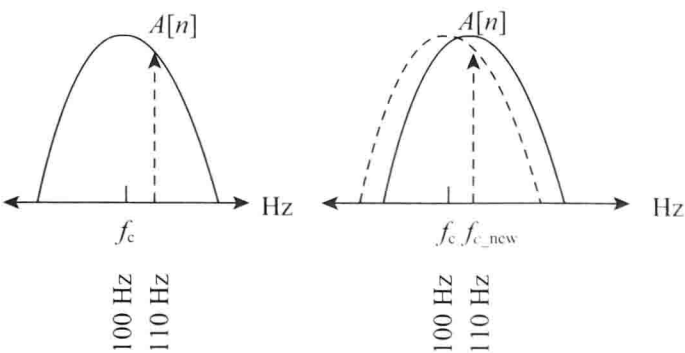


图 3.20 二次谐波的真实频率位置
(通道中心频率 $f_c = 100\text{Hz}$,再合成后中心频率 f_{c_new} 与瞬时频率有偏移)

如前所述,通道声码器的滤波带有固定的中心频率(即带通滤波器组的中心频率),输入信号被分解成若干个子带,每个滤波带只计算该子带的能级(瞬态幅值 $A[n]$)。与编码端一样,解码端也有着中心频率相同的固定滤波带。对于相位声码器,情况则不同。

设输入为一个复合谐波信号,由基频 55Hz 和其前三次谐波组成。该例中,八度(相当于二次谐波)将位于 110Hz。设出于某种原因,滤波带被校准后,使二次谐波(110Hz)进入了中心频率为 100Hz 的滤波带,如图 3.20 的左图所示。由图可知,该子带的输入为二次谐波 110Hz,与该滤波带的固定中心频率 100Hz 不一致。问题随之产生了:合成模块(解码端)的解码器不能判断(或者说对于通道声码器,并不关注)原信号中对应的频率分量是多少。因此,在合成过程中,中心频率为 100Hz 的子带共振频率仍为 100Hz 而不是 110Hz,而由包络跟随器计算的瞬态幅值 $A[n]_{100\text{Hz}}$ 所控制的近似能级保持不变。

所以,相位声码器优于通道声码器的做法是,在编码模块的特定通道对瞬时频率进行估计,并作为合成时更精确的子带输入中心频率。则在合成过程中,解码器将以所分析的瞬时频率作为加法合成技术的中心频率重构输出信号。因此,与通道声码器不同,相位声码器直接将带有适当瞬时幅值和瞬时频率的正弦波进行叠加以合成输出信号,该声码器不再基于信源滤波器模型。那么,如何计算瞬时频率呢?请见下节。

3.4.1 瞬时频率估计

第4章第2节已简要介绍了瞬时频率的定义,如式 3.20,其中 $\Theta(t)$ 为正弦波(式 3.21)的相位[此处为 $\omega t + \phi$, 不要与 3.22 节正弦波中的初相位 ϕ 相混淆]。式 3.22 的初始相位 ϕ 也可以是时间的函数 $\phi(t)$ 。当 ϕ 是一个只表示初始相位的常数时,则 $\phi(t)$ 当然也是一个常数。

$$f(t) = \frac{1}{2 \cdot \pi} \frac{d\Theta(t)}{dt} \quad (3.20)$$

$$y(t) = \sin\Theta(t) \quad (3.21)$$

$$\Theta(t) = 2 \cdot \pi \cdot f \cdot t + \phi = \omega \cdot t + \phi \quad (3.22)$$

数字系统中,瞬时频率通常由两个相邻的 STFT 帧(使用傅立叶变换)计算,首先计算第 k 个频率帧的相位值,如式 3.24,然后计算各自相角展开角的差,最后除以两个连续 STFT 帧 $X^d[\cdot]$ 的帧长 R ,如式 3.26,其中除数 2π 是一个归一化系数,使结果 f 的单位为 Hz 而不是弧度。计算结果为下一时刻 $m+1$ 的瞬时频率 f_{m+1} 。

$$X^d[k] = a + jb \quad (3.23)$$

$$\angle X^d[k] = \varphi[k] = \tan^{-1} \frac{b}{a} \quad (3.24)$$

$$|X^d[k]| = \sqrt{a^2 + b^2} \quad (3.25)$$

$$f_{m+1} = \frac{1}{2\pi} \cdot \frac{\varphi_{m+1} - \varphi_m}{R} \cdot f_s \quad (3.26)$$

如果考虑相位在单位圆上的变化以及相位本身在单位圆内的定义,则能更好地理解瞬时频率的含义。以正弦曲线为例,如图 3.21 所示,它可看成由逆时针方向沿单位圆运动、同时在 x 轴上匀速运动的点所经过的轨迹,移动的时间间隔或者说时钟“滴答”声间隔为 $1/f_s$ 。由于频率以“每秒”为单位,因此瞬时频率由运动所经过的两个连续点的相位值除以时间间隔而得到。此例时间间隔 $T=1/f_s$ 秒,每个采样点的相位由式 3.24 计算。相位的思想如图 3.21 所示:

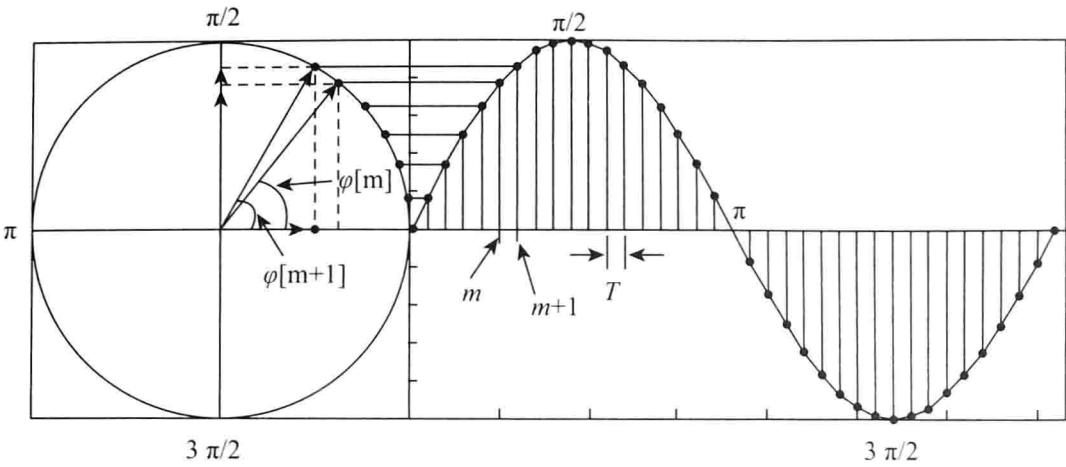


图 3.21 瞬时频率与相伴变化率

因此,若采用中心频率为 100Hz 的带通滤波器,输出以 $a+jb$ 的复数形式表示,可按上述思路计算其瞬时相位和瞬时频率(将图 3.21 中的 y 轴做为虚轴)。对于处在两个连续离散时间点 $n=m$ 和 $n=m+1$ 上的采样点,可计算其瞬时频率如式 3.26,样本时间间隔 $R=T$ 。为了获得正确的瞬时频率值,必须进行相位展开,因为弧度相位具有 2π 模数特征(相位大于一个完整周期时,所计算的相位差不连续),详细讨论见 3.4.4 节。

3.4.2 相位展开

计算瞬时频率之前,需要进行相位展开(phase unwrapping)。第 6 章第 3.3 节对相位展开已经有过详细介绍,需要注意的是,相位超过 2π 弧度时,会以 2π 递增(即 2π 模数系统)。因此,进行相位展开的原因是,计算两个相邻采样点的相位差 $\varphi_2-\varphi_1$ 时,因 2π 模数特征将有可能得到错误的瞬时相位差(而为了计算瞬时频率需要获得相位的变化量)。为了阐明这一事实,以计算中心频率 $f_c=100\text{Hz}$ 的滤波带的瞬时相位为例,计算所得的相位序列 φ_{nT} 如式 3.27,其中 n 为采样点序号,采样周期 $T=1$ 秒(为了方便,该例的相位单位是角度而不是弧度)。

$$\varphi_1 = 0, \varphi_2 = 95, \varphi_3 = 190, \varphi_4 = 285, \varphi_5 = 10, \varphi_6 = 105$$

(3.27)

然后计算上述六个相位值的瞬时频率如下：

$$\frac{\varphi_1 - \varphi_2}{T} = \frac{95 - 0}{1} = 95$$
$$\frac{\varphi_3 - \varphi_2}{T} = \frac{190 - 95}{1} = 95$$
$$\frac{\varphi_4 - \varphi_3}{T} = \frac{285 - 190}{1} = 95$$
$$\frac{\varphi_5 - \varphi_4}{T} = \frac{10 - 285}{1} = -275$$
$$\frac{\varphi_6 - \varphi_5}{T} = \frac{105 - 10}{1} = 95$$

(3.28)

由上式可知,瞬时频率似乎固定在 95Hz,而到了第 4 和第 5 帧之间:瞬时频率为 -275Hz。

这里看起来出了点问题,因为这种频率的剧烈变化是不正常的,或者说理想正弦波的频率是不可能发生变化的。这是由 360° (或 2π) 模数的相位包裹所产生的典型情况——相位超过 360° 或 2π 时的初始化有点像手表指针超过 12 点时的情形。因此,对第 5 帧进行相位展开——即加上一个 360° (或 2π) 就能够得到正确的瞬时频率:

$$\begin{aligned}\frac{\varphi_5 - \varphi_4}{T} &= \frac{(10 + 360) - 285}{1} = 95 \\ \frac{\varphi_6 - \varphi_5}{T} &= \frac{(105 + 360) - (10 + 360)}{1} = 95\end{aligned}\quad (3.29)$$

图 3.22 给出了一个正弦波相位展开图,其中直线为展开后的相位,锯齿状曲线为展开前的相位。

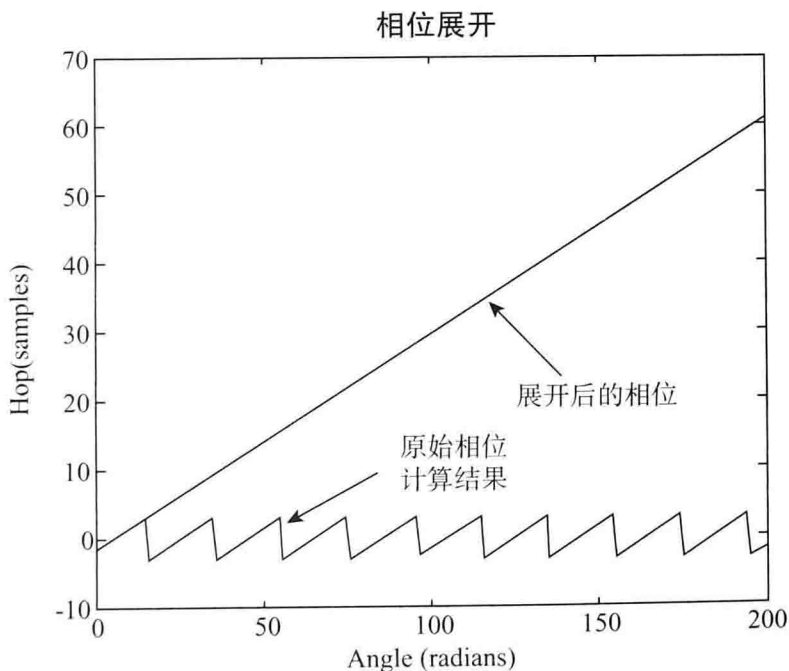


图 3.22 相位展开

3.4.3 相位声码器:滤波带组模型

到目前为止,对于相位声码器机制的解释,最完美的是“滤波带组模型”(filter-bank interpretation, Dolson 1986),它将相位声码器看成是编码和解码端的一组带通滤波器。本节将用该理论来解释相位声码器的机制,并通过这一过程来阐述获得瞬时频率的方法。如前所述,分析模块(编码器)对瞬时幅值和瞬时频率进行估计,而合成模块(解码器)则以这两个参数通过加法合成技术重构信号——即将具有各自频率分量的正弦波进行叠加,如图 3.23 和图 3.24,其中 $A[m]$ 和 $f[m]$ 分别为特定滤波带的瞬时幅值和瞬时频率。

接下来从滤波带模型的角度详细讨论以瞬时频率估计来计算瞬时相位的方法。首先将问题简化为编码器的瞬时频率计算模块,相位声码器中一个通道的信号流程图如图 3.25 所示(设每个通道除了中心频率外其他设置都相同)。该流程图看起来可能相当复杂,也不像第一眼看上去那样简单,但如果将整个过程按步骤分解为不同的子模块,处理过程将变得清晰,并且比实际看上去要简单。至此,我们的目标仍然是以两个相邻时间帧的瞬时相位计算来估计瞬时频率。

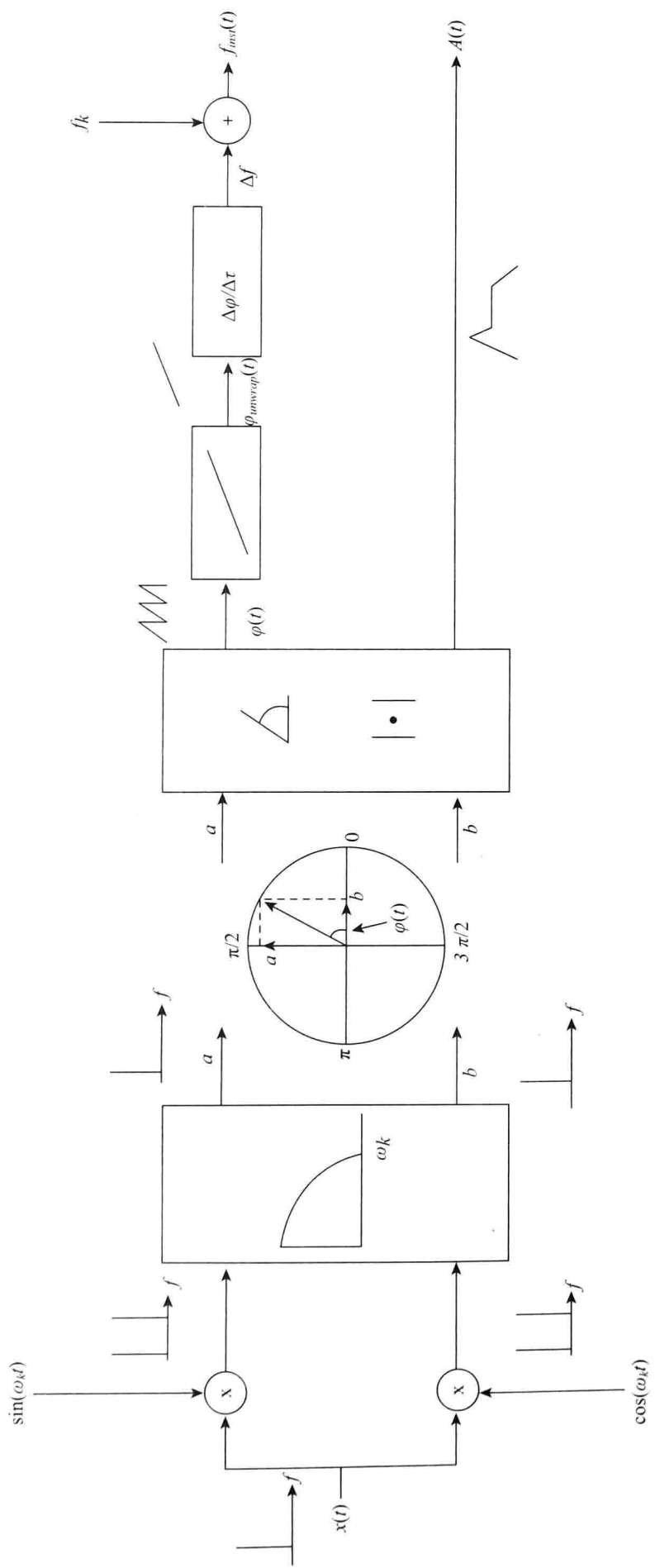


图3.25 由滤波带法生成的相位声码器

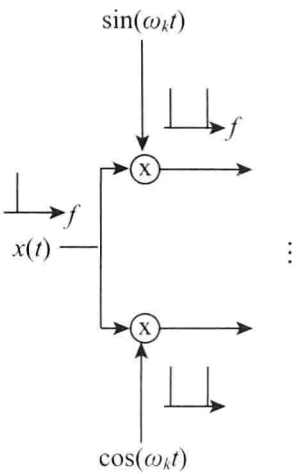


图 3.26 外差计算

进行 DFT 变换时,每帧的输出如式 3.30,可用式 3.31 和 3.32 计算它的相位以及幅值。但如果不使用傅立叶变换,如何计算幅值和相位呢? 答案是使用外差计算。

$$c = a + jb \tag{3.30}$$

$$\angle c = \varphi = \tan^{-1}\left(\frac{b}{a}\right) \tag{3.31}$$

$$|c| = \sqrt{a^2 + b^2} \tag{3.32}$$

外差计算提供了一种便利的相位计算方法,它可将输入信号 $x(t)$ 改写为直角坐标系正弦和余弦的乘积形式(就像傅立叶变换的实部与虚部相加的复数坐标),如图 3.26。那么外差计算是如何将结果转化到直角坐标系的呢? 调幅和拍频的相关知识(第 4 章第 4.1 节)可以回答这个问题。考虑如下的三角恒等式:

$$\cos(A) \cdot \cos(B) = \frac{\cos(A - B) + \cos(A + B)}{2} \tag{3.33}$$

上式的和差基本关系($A+B$ 与 $A-B$)构成了外差计算的基础。在外差计算中,输入信号与 $\sin(\omega t)$ 和 $\cos(\omega t)$ 的乘积形成了一个和差对。和分量被低通滤波器去除,而差分量($A-B$)则会保留。因为余弦分量的相位总是提前 90° ,所以余弦分量和正弦分量的相差永远是 90° 。因此外差计算的输出——即经过乘法和低通滤波后,正弦和余弦分量的频率以($A-B$)频移至直流分量,每个分量之间存在 90° 的相位差。此例频移量 $f_k = 100\text{Hz}$ 。两个分量(输入以余弦和正弦进行外差计算后的输出)之间的 90° 相差经低通滤波后形成了一个直角坐标系,如图 3.27 所示。

由输入信号的这种变换形式得到的输出分量(垂直和水平分量)称为同相正交分量(in-phase and quadrature components)。既然已经得到了直角坐标系统,则可由式 3.31 计算输入信号的相位,由式 3.32 计算幅值,如图 3.28。

如前所述,相位声码器本身没有“滤波带组”(即传统的带通滤波器),但利用外差计算过程可实现与带通滤波器类似的功能。对瞬时幅值和瞬时相位的计算而言,这一做法是可行的。下一步是计算瞬时频率,或者更准确地说是计算“外差处理”后的瞬时频率,因为在外差计算过程中,输入载波信号已经由 f_k 频移到了直流分量中。该值可由两相邻的瞬时相位值得到,后面将

进一步论述。为了计算瞬时频率,下一步需要进行相位展开(相位展开的原因详见 3.4.2 节),如图 3.29,展开后的相位为 $\varphi(t)_{unwrap}$ 。

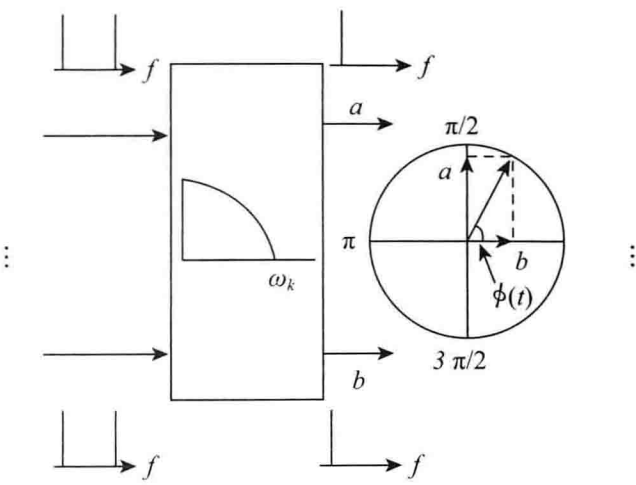


图 3.27 有外差输出的低通滤波器

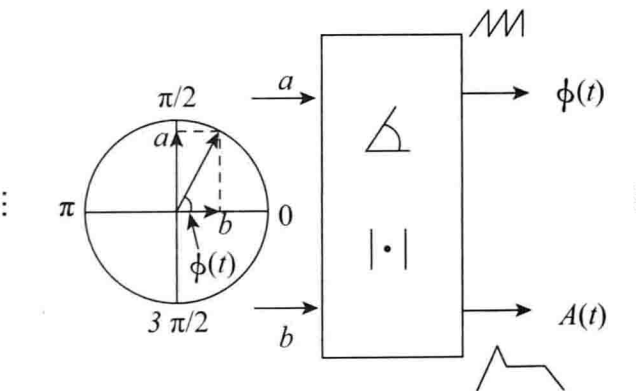


图 3.28 相位与幅值包络的计算

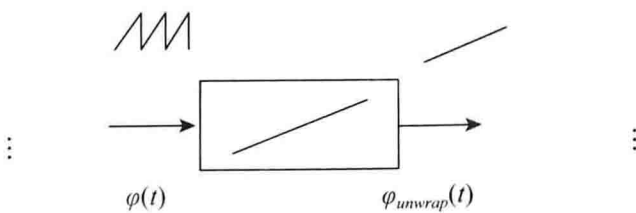


图 3.29 相位展开

相位展开的过程由两个时间间隔为 $\Delta\tau$ 的相邻帧完成,则瞬时频率由两帧间相差 $\Delta\varphi$ 除以 $\Delta\tau$ 相除而得到。将信号送入解码器前最后需要做的是,将瞬时频率转化至正确的频率范围。由于外差计算,输入被平移至了 DC 直流分量,频移量为 f_k ,因此最后需要向外差瞬时频率添频移补偿量 f_k 。例如,输入的正弦波频率为 102.34Hz,外差计算将该分量频移至了 2.34Hz,需要将外差瞬时频率 Δf 加上频移量 f_k 以得到真实的瞬时频率,如图 3.30:

上述内容从滤波带组理论的角度总结了相位声码器的编码端机制,如图 3.25 所示。到此,我们希望图 3.25 看起来不如之前那么复杂。对于解码端,生活也许往往就如平常一样简单,只需输入频率和幅值参数,并对带有编码器传送参数的正弦分量进行叠加,然后再重构输出信号。

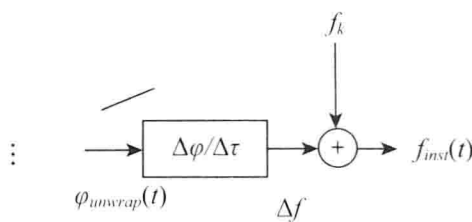


图 3.30 引入后级中心频率 f_k 来计算瞬时频率

3.4.4 相位声码器:傅立叶变换模型

从傅立叶变换的角度看,相位声码器的构架基于输入信号的时窗重叠 STFT 变换。在本质上,相位声码器的傅立叶变换模型与理论滤波带组模型非常相似(也应该相似)。相位声码器的 STFT 模型也包含一个与“滤波带组”结构相适应的“带通滤波器组”,并均匀分布于离散频率帧上——每个频率帧 $k(f_s/N\text{Hz}, N$ 为 DFT 变换长度)可看做对应着一个通道。例如, N 为 1.024 个采样点,采样频率 $f_s=8\text{ kHz}$,频率分辨率则为 7.8125Hz。对于第 1 至第 3 个频率帧,中心频率分别对应为 7.8125Hz、15.625Hz 和 23.4375Hz。

$$X^d[k] = \sum_{n=0}^{n=N-1} x[n] \cdot e^{-j\frac{2\pi kn}{N}}$$

(3.34)

在相位声码器的滤波带组模型中,利用外差计算获得正交分量以实现瞬时相位的计算。在傅立叶变换模型中,相位可直接由频率帧本身计算,瞬时相位和瞬时频率则由两个相邻 STFT 频率帧计算。由式 3.34 可知,DFT 变换也包含类似外差计算的乘法过程——即输入信号 $x[n]$ 与由余弦和正弦分量组成的欧拉公式项的乘积。

相位声码器可灵活用于时间延展(time-stretching)和移调(pitch-shifting),并同时可避免由减采样带来的“芒奇金效应”或由过采样带来的“逆芒奇金效应”。如前所述,从频域的角度看,在相位声码器的傅立叶变换模型中,分析模块的输出可看成滤波带——每个带由各自的频率、幅值和相位参数表示。合成模块提取这三个参数(也许是经过某种调制的),并重构最后的输出信号,最终通过 ISTFT 变换重组时域信号——这相当于重叠 STFT 频率帧的叠加。上述基本思想如图 3.31,其中 STFT 模块承担分析任务,谱分析处理模块进行频谱的调制、转换与调整,最

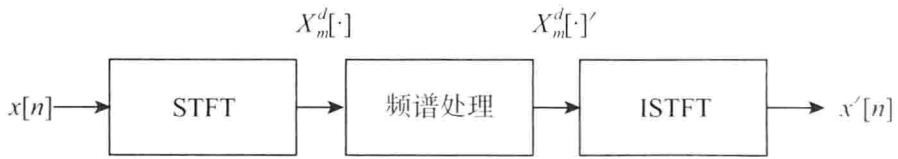


图 3.31 相位语音合成的基本结构

后 ISTFT 模块负责正确地将信号还原为时域信号。

我们已经讨论了计算瞬时频率之前进行相位展开的必要性,毋庸置疑,傅立叶变换模型也需要对每个频率帧 k 进行相位展开。为了获得两个相邻 STFT 频率帧之间的展开瞬时相位变化,并用相位展开算法计算瞬时频率,该处理过程的最终目标是保留瞬时频率信息。在相位声码器的 STFT 模型中,每个频率帧 k 对应着一个通道,通道以离散频率按逆时针方向和特定角速度旋转,就像一个旋转的车轮。频率较低的帧旋转得慢(DC 直流分量不旋转),频率较高的帧旋转得快——该频率帧 k 的“弧度距离”(radian distance)可由式 3.35 计算,其中 N 为频率轴的步进(即时窗宽度或 DFT 变换长度), k 为帧序号, φ 为频率帧 k 的相位。

$$\varphi[k] = \frac{2k\pi}{N} \quad (3.35)$$

由上式可知,对于频率较高的帧,周期数(旋转完全一周)比频率较低的帧大。就好比兔子、浣熊和龟在环形赛道赛跑,测量每 5 分钟各自跑过的距离,就是这个道理。下一帧的展开瞬时相位由当前帧提取的相位值(旋转 2π 弧度的数量)计算,并用所获得的下一帧的实际相位对当前帧进行修正,如图 3.32,为了便于理解相位声码器 STFT 模型的相位展开概念,图中相位展开后的单位为度($^{\circ}$)而不是弧度。

例如,设当前帧 s 和下一帧 $s+1$ 之间的 STFT 分析跳距为 R_a 个采样点。如果未知频率帧 k 的角频率为 45° 每采样,在 R_a 个采样点,下一帧 $s+1$ 的理论瞬时相位值为 45° 的 R_a 倍。也就是说,如果完整的正弦波形没有被分析时窗从中间截断(时窗化的片断是完整正弦波形),则所提取的瞬时频率的计算结果如式 3.36,相位展开如图 3.32:

$$\varphi_{(s+1)R_a}^p[k] = \frac{360 \cdot k}{N} R_a + \varphi_{sR_a} \quad (3.36)$$

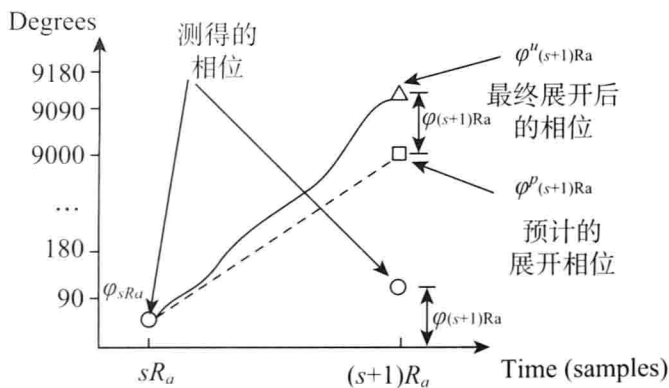


图 3.32 两个连续 STFT 帧的相位展开(基于 Zölner 2002)

但如果信号按准正弦周期变化,图像如图 3.32 中曲线而不是虚线,则瞬时相位就无法再用式 3.36 进行计算,因为当前帧 s 和下一帧 $s+1$ 之间的相位变化关系未知。因此,在一般情况下,瞬时相位的计算需要已知当前帧 s 经过了多少个周期(即 360 或 2π 的乘积因子 m 为整数)才能到达未知帧 k 的角频率(此例为 45°),然后加上帧 $s+1$ 的实测相位(由 DFT 变换获得),最后结果如式 3.37 所示:

$$\varphi_{(s+1)R_a}^u[k] = 360 \cdot m + \varphi_{(s+1)R_a}^p[k] \quad (3.37)$$

整数 m 可由式 3.38 求得。由式 3.38 可计算式 3.36 的 360° 的整数分辨率。

$$m = \text{int}\left(\frac{\varphi_{(s+1)R_a}^p[k]}{360}\right) \quad (3.38)$$

例如,设给定频率帧 $k=32$,每采样的角频率更新率为 45° ,时窗宽度 $N=256$,当前帧 s 的相位 $\varphi_s=47^{\circ}$,跳距 $R_a=200$,则所提取的该频率帧的展开瞬时相位和模数 360 的乘积因子 m 的计算结果如下:

$$\begin{aligned}\varphi_{(s+1)R_a}^p[k] &= \frac{360 \cdot k}{N}R_a + \varphi_s[k] \\ &= \frac{360 \cdot 32}{226}200 + 47 \\ &= 45 \cdot 200 + 47 \\ &= 9047^\circ\end{aligned}\tag{3.39}$$

$$\begin{aligned}m &= \text{int}\left(\frac{\varphi_{(s+1)R_a}^p[k]}{360}\right) = \text{int}\left(\frac{9047}{360}\right) \\ &= \text{int}(25.13056) \\ &= 25\end{aligned}\tag{3.40}$$

由式 3.40 可知,可用式 3.37 计算最后的展开瞬时相位,且在此例中,对于频率帧 $k=32$,通过复数表示的实测相位 $\varphi_{(s+1)R_a}$ 为 120° 。该展开瞬时相位的计算结果为 $9,120^\circ$,如式 3.41 所示,相位展开如图 3.32 所示。同时,以上计算各阶段结果在圆上的表示如图 3.33——最内侧的圆周为 $m=0$ 时的相位,其外侧的圆周为 360° 附加相位,最外侧的圆周则为由式 3.37 计算所得的 $m=25$ 时的相位。

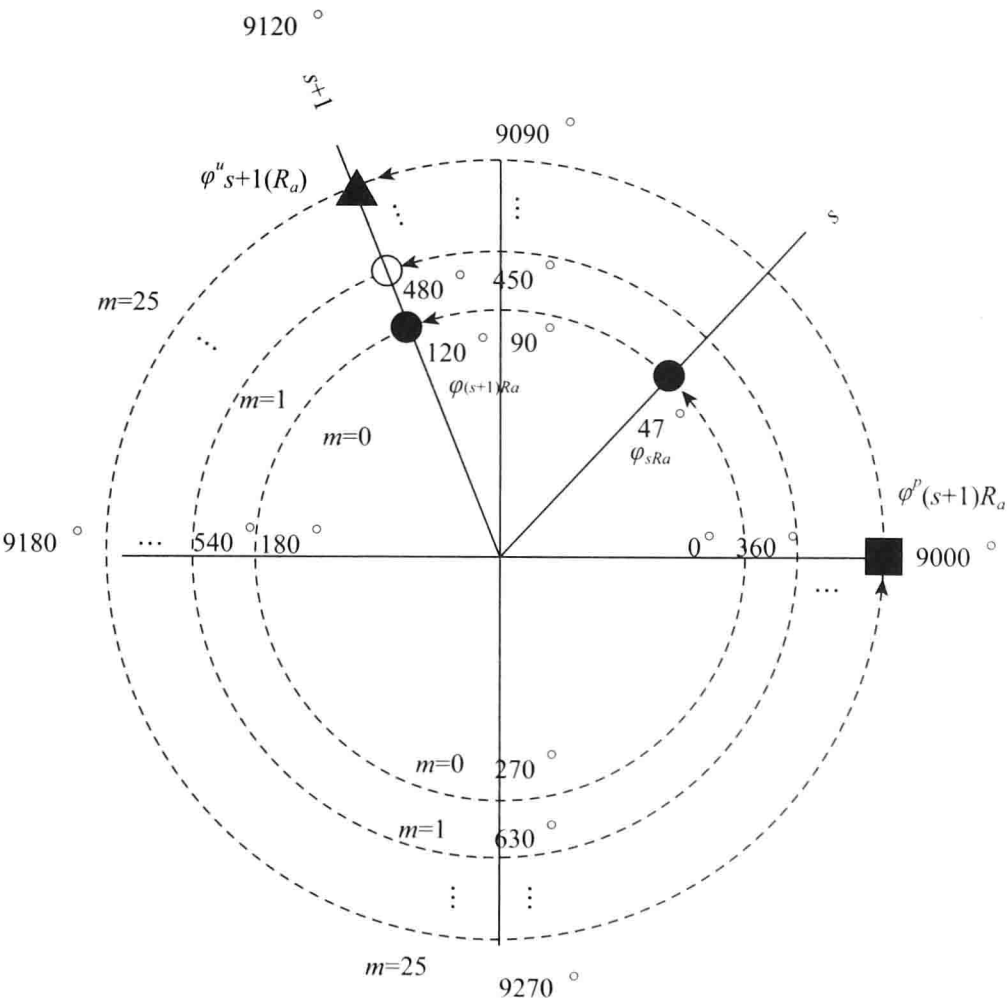


图 3.33 通过计算瞬态相位进行相位展开

$$\begin{aligned}
\varphi_{(s+1)R_a}^u[k] \big|_{k=32} &= 360 \cdot m + \varphi_{(s+1)R_a}[k] \\
&= 360 \cdot 25 + 120 \\
&= 9120
\end{aligned} \tag{3.41}$$

相位展开时,以角度而不是弧度为单位会使结果更加清晰,不过相位展开式当然也可以以弧度为单位,则式 3.36、3.37 和 3.38 可改写为式 3.42、3.43 和 3.44,上例的展开结果为 $9120 \cdot \pi/180 = 50.67 \cdot \pi$ 弧度。

$$\varphi_{(s+1)R_a}^p[k] = \frac{2\pi \cdot k}{N} R_a + \varphi_s[k] \tag{3.42}$$

$$\varphi_{(s+1)R_a}^u[k] = 2\pi \cdot m + \varphi_{(s+1)R_a}[k] \tag{3.43}$$

$$m = \text{int}\left(\frac{\varphi_{(s+1)R_a}^p[k]}{2\pi}\right) \tag{3.44}$$

由式 3.44 可知,用式 3.45 可计算出两个相邻帧 s 和 $s+1$ 之间的展开相位差。然后用该相位差可由式 3.26 计算瞬时频率,如式 3.46,将上述计算结果与采样频率相乘可得单位为 Hz 的瞬时频率。

$$\Delta\varphi_{(s+1)R_a}[k] = \varphi_{(s+1)R_a}^u[k] - \varphi_{sR_a}[k] \tag{3.45}$$

$$f_{(s+1)}[k] = \frac{1}{2\pi} \cdot \frac{\Delta\varphi_{(s+1)R_a}[k]}{R_a} \cdot f_s \tag{3.46}$$

3.4.5 相位声码器的基本应用:移时与移调

第 2 章简要介绍了在时域内,在分析端和合成端利用不同的跳距,通过 OLA 叠加算法和时窗分析法(这些算法不关注瞬时相位和瞬时频率)进行时间延展和压缩。也就是说,分析端的跳距 R_a 小于合成端的跳距 R_s 时,可获得比原信号更长的信号,这一过程称为时间延展,反之亦然。第 2 章也提到了一种效果不佳的时域移调算法,利用该算法进行移调将改变信号的时长,不经意间会产生副产品,即“芒奇金效应”。而现在,利用相位声码器可更好地进行移时和移调,并且几乎不增加人为失真(如果介意的话)。本节将讨论与音频信号的移时和移调有关的问题和基本算法。

3.4.5.1 移时

通过相位声码器进行移时和移调有两种主要方法(滤波带模型和 ISTFT 变换模型),两者的分析模块都同样基于 STFT 变换,但合成模块不同。在分析模块中使用 STFT 变换时,可在合成端选择滤波带模型(正弦波的叠加)或 ISTFT 变换模型。接下来首先讨论滤波带模型。

移时的滤波带模型:上一节已经讨论了分析模块和瞬时频率的计算。对于合成模块,采用滤波带模型,移时的效果可分别用插值或抽取算法延长或缩短正弦分量实现。对于时间延展,需要在每个正弦分量(频率帧 k)的当前帧和下一帧之间插入一个“新”帧的幅值,对于时间压缩则进行帧抽取。一个时间延展的实例如图 3.34,跳距 $R_s = 3 \cdot R_a$,移时比定义为式 3.47。

$$\text{time_shift_ratio} = \frac{R_s}{R_a} \tag{3.47}$$

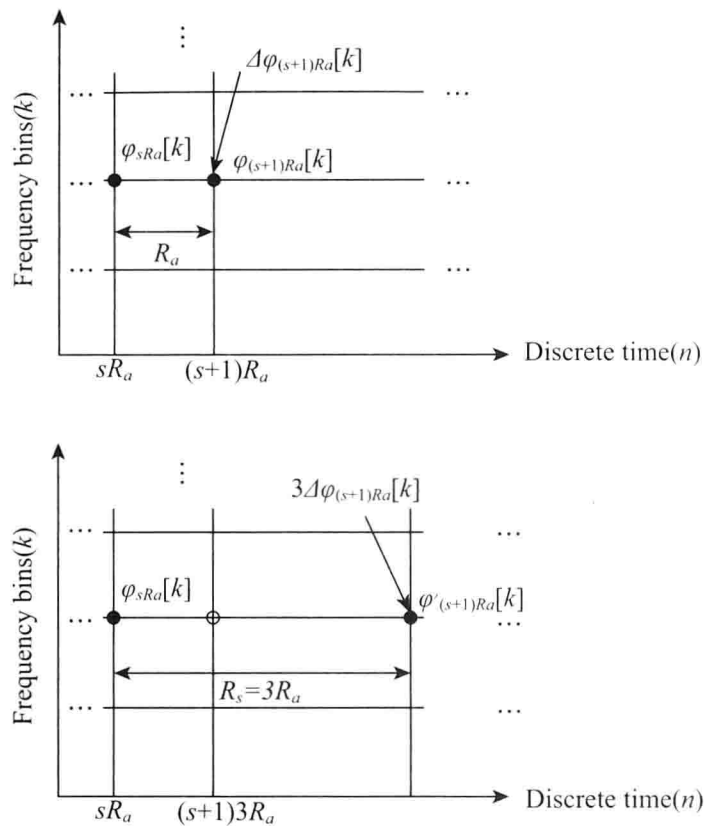


图 3.34 3 倍时间拉伸, $R_s=3 R_a$, 上图为分析结构, 下图为延展后的结果

时域时间延展算法(见第 2 章)和频域时间延展算法的重要区别是,在离散时间轴上合成给定信号时,频域算法在控制各频率分量及其相位特征的同时,可保持信号的瞬时频率特性不变。第 2 章介绍的过采样或减采样算法无法实现这一点。已知时间延展前原位置的两个相邻帧 s 和 $s+1$ 之间的瞬时相位差 $\Delta \varphi_{s+1}[k]$,则可计算插入一个新帧所需的相位增量 $(\psi_{(s+1) R_a}[k])$,从而求得当前帧与该新帧的瞬时相位 $\varphi'_{(s+1) R_a}[k]$,如式 3.48。

$$\varphi_{(s+1) R_a}[k] = \frac{\Delta \varphi_{(s+1) R_a}[k]}{R_a} \tag{3.48}$$

在原始帧和时间延展帧(n 由第 $s \cdot R_a+1$ 个采样点开始至第 $3 \cdot s \cdot R_a$ 个采样点结束)之间新插入的相位值(由上标 i 表示)由式 3.49 计算。每个采样点 n 上的合成样本 $y[n]$ 最终由 $N/2$ 个有着对应幅值和瞬时相位的正弦分量重构(N 为时窗宽度或 FFT 长度, k 为帧序数),如式 3.50。

$$\varphi_{n+1}^i[k] = \varphi_n[k] + \psi_{(s+1) R_a}[k] \tag{3.49}$$

$$y[n] = \sum_{k=0}^{N/2} A_n[k] \cdot \cos(\varphi_n^i[k]) \tag{3.50}$$

图 3.35 给出了时间延展过程中幅值和相位的更新过程,同时由该图可知瞬时频率对于正弦波重构和最终音质的重要性。

与图 3.34 的下图相同,图 3.35 也给出了新增幅值插值,并用当前帧与新插入帧 $(s+1) \cdot 3 \cdot R_a$ 之间的相位增量或插值量表示。由图可知,位于帧 k 的正弦幅值分量是由 $A_{s R_a}[k]$ 至 $A'_{(s+1) R_a}[k]$ 的线性插值,在该时刻(n)为该正弦分量提供了 $s R_a$ 和 $(s+1) 3 R_a$ 之间的正确幅值。

对于当前帧与下一帧的相位插值,相位增量为 $\psi_{(s+1)Ra}$,如式 3.48,这一增量保持了瞬时频率——在这两帧内,式 3.48 的弧度增量(即每采样移动的“弧度距离”)得到了维持。因此,在得到了每个采样点的正确幅值插值 $A_n[k]$ 和瞬时相位插值 φ_{n+1}^i 后,可用式 3.50 通过对 $N/2$ 个正弦分量(DC 直流分量至奈奎斯特限制频率分量)的叠加计算每个采样点的合成输出。每个时刻 n 对应的采样点用 $N/2$ 个正弦分量表示。

对于时间压缩,只需进行抽样处理就能减少当前频率帧 s 和下一帧 $s+1$ 的帧之间的采样点数量。而与时间延展有关的公式如图 3.36 所示,其中时间压缩后的信号与原信号的长度比为 30%。

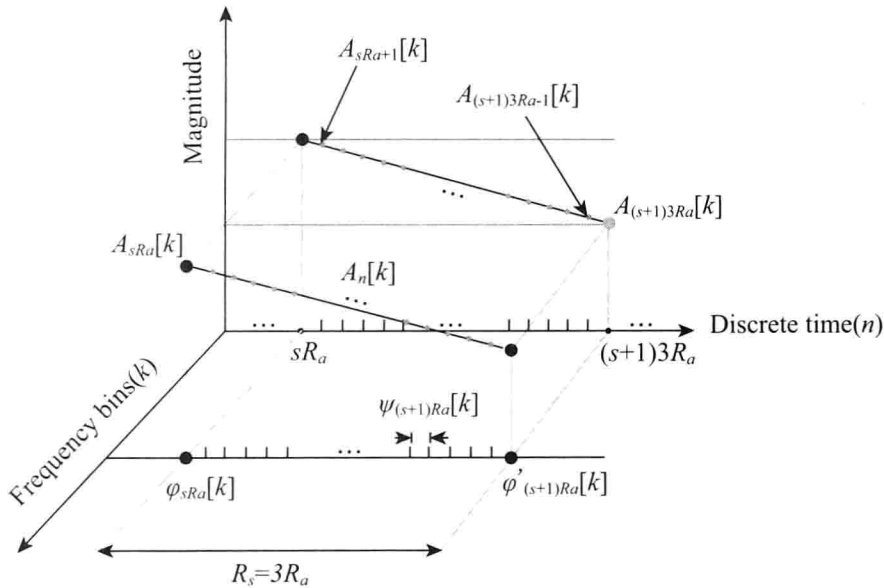


图 3.35 3 倍时间拉伸后的幅值与相位, $R_s = 3R_a$

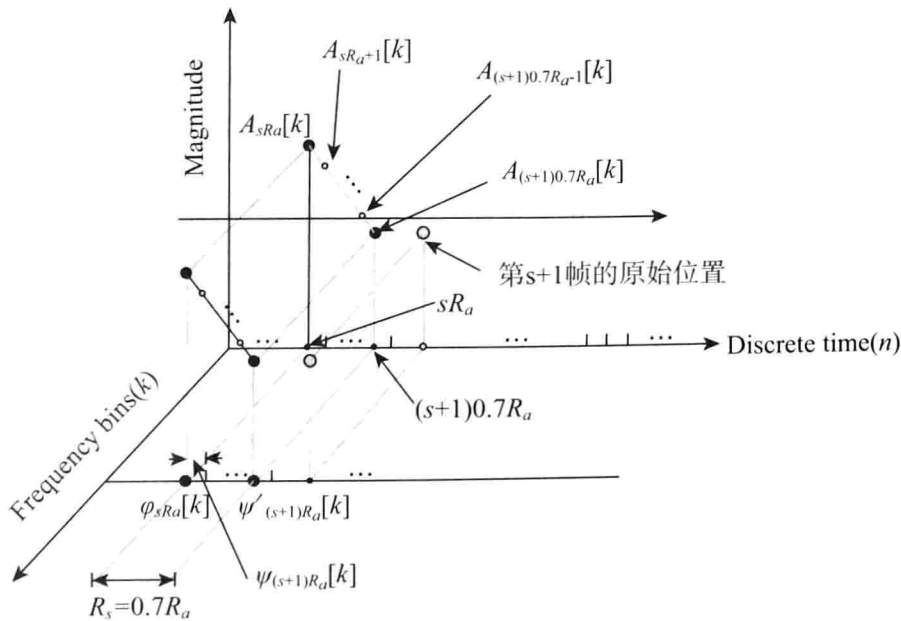


图 3.36 时间压缩后的幅值与相位, $R_s = 0.7R_a$

移时的 ISTFT 模型:移时相位声码器的 ISTFT(重叠 DFT 变换)模型与上一模型相似。在傅立叶变换中,合成跳距与分析跳距的比值 R_s/R_a 用于确定延展或压缩因子,同时保留分析

模块计算的瞬时频率特征。但与滤波带模型不同的是,逆傅立叶变换模型是逐帧重构而不是逐个采样点重构,通过 ISTFT 变换获得的每个 IDFT 变换结果对应的是一个由合成跳距 R_s 和与 DFT 长度所决定的时间段,如图 3.37 所示。在滤波带模型中,每个时刻 n 对应的新输出样点由一系列正弦分量组成,而在 ISTFT 模型中,每个 DFT 频率帧(包含各自的 $N/2$ 个正弦分量)的长度与 DFT 时窗长度相同,然后在合成端通过 OLA 叠加算法产生移时的输出信号。

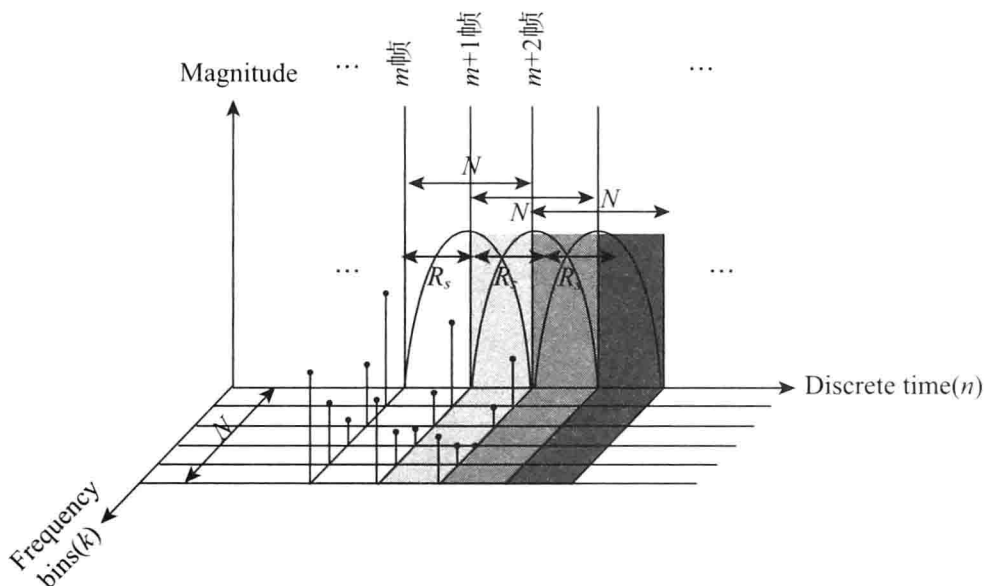


图 3.37 移时的 ISTFT 模型的分析模块

如前所述,移时的主要问题是在合成端用新的展开伸缩相位增量 $\Delta\varphi'_{(s+1)R_s}$ 来维持瞬时相位,如式 3.51:

$$\Delta\varphi'_{(s+1)R_s}[k] = \frac{R_s}{R_a} \cdot \Delta\varphi_{(s+1)R_s}[k] \quad (3.51)$$

随着新相位增量的增加,瞬时频率得到维持,因此跳距由比值 R_s/R_a 决定。而该新增的相位增量也需要加到之前的展开相位 φ'_{sR_s} 中,以保持相位的同步增长,如式 3.52:

$$\varphi'_{(s+1)R_s}[k] = \varphi'_{sR_s}[k] + \Delta\varphi'_{(s+1)R_s}[k] \quad (3.52)$$

最后可用式 3.52 的结果和更新后的相位值计算修正后的 DFT 帧 k 的分量 $X^d[k]'$,如式 3.53,然后再由前后连续的 ISTFT 帧的 OLA 叠加算法重构时域信号。

$$X^d[k]'_{(s+1)R_s} = (|X[k]_{(s+1)R_s}|) \cdot e^{j\varphi'_{(s+1)R_s}} \quad (3.53)$$

时间压缩的 ISTFT 模型与之前的时间延展过程一致,不过会用相位衰减量来代替相位增量。移时合成的 ISTFT 模型如图 3.38:

3.4.5.2 移调

移调(pitch shifting)是频率分量与转移因子的乘积(而不是增加偏移量),在移时算法之后再重采样就能进行移调,如图 3.39 所示(Dolson 1986, Zölner 2002)。例如,要将信号向上频移一个八度,可先通过相位声码器进行两倍时间延展,然后按 $1/2$ 的比率对延展后的信号进

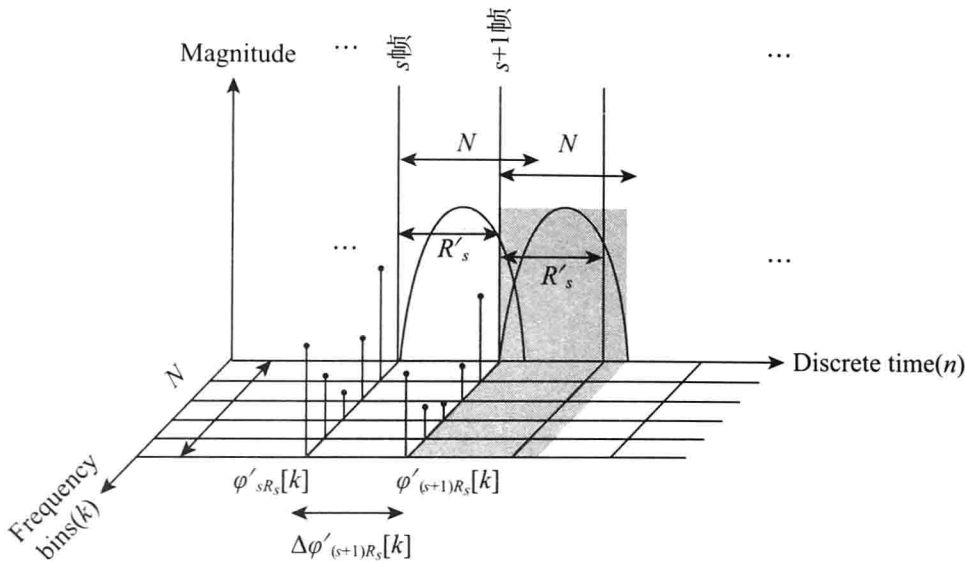


图 3.38 用于分析时间拉伸的 ISTFT 模型

行重采样。这样处理的原因是,用相位声码器进行时间延展时不会发生移调,但得到了比原始信号多一倍的采样点。然后再进行两倍减采样时,音高将增加一个八度,而信号也将还原为原始信号的长度,因为进行两倍时间延展后再以同样的比率进行减采样将得到原始信号长度。

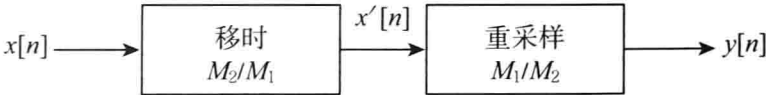


图 3.39 通过时间延展与重采样进行移频

有许多方法可用于增强和优化相位声码器,例如,当时窗重叠量的平方和等于一个使 $R_s = R_u$ 的优化值时,可获得精确的重构信号。在此,本书不再过多地讨论相位声码器的改进算法和细节,不过希望本节所介绍的基本概念能够启发读者踏上研究“相位声码增强器”(phase-vocoolightenment)的漫长旅程。

4. 计算机音乐的若干研究专题

可以想像,如今在计算机音乐领域正进行着一系列深入的研究和探索,尤其是功能日益强大的个人电脑为该领域的研究展现了广阔的应用前景。多核处理器技术为模型的仿真、算法概念的测试、旧算法在实时系统中的应用以及所有声音合成与分析领域的研究提供了良好的硬件环境。最后,我们将用一些最近引人注目且令人兴奋的计算机音乐研究专题为全书做一个总结。

4.1 显著特征量的提取

近来在计算机音乐研究中备受关注的领域之一是音乐信息检索(MIR, music information retrieval)。这一跨学科领域的研究者主要关注于提取音乐和声音的检索信息。该类检索的应用实例包括音乐流派风格的自动分类、歌曲哼唱查询系统、节奏检测和自动乐器音色分类等。下节将介绍一些音乐信息检索 MIR 的研究专题并在本节讨论音乐信号的显著特征提取(salient

feature extraction)算法,它是普通音乐模式识别和 MIR 的重要子研究领域。

显著特征量提取关注于信号中重要信息、特征和参数提取算法的研究与开发——在我们关注的领域中,这里的信号主要指音频信号。例如,在读者面前快速地展示一支笔,时间约为一秒,然后要求读者识别该物体,读者可能很快就会告诉我们这是某个类型的笔。实际上人们对物体的反应速度是相当惊人的——尽管事先没有被告这是一只自动铅笔、钢笔或圆珠笔。同理,对于熟悉的乐器声,如小号或吉他——即使只听到一小段稍纵即逝的乐音,也有可能很快说出该乐器的名称,甚至音高。这种识别能力很大程度上来自于人们对物体突出特征的判断。对于笔的例子,由经验可知,大小一定的笔为圆柱形或其他类似形状,同时通过大小本身的特征(粗或细),还可以判断这是一支细的书写笔还是粗的马克笔。从某种意义上说,这种认知方法降低了描述事物的维度或在大量信息中找到了重要而显著的信息。其中一种寻找显著特征的方法是寻找以下一系列问题的答案——如:“哪些基本因素和参数有助于识别某个物体?”“哪些参数集或特征向量足以反映或表示某个物体?”本节将介绍一些与声音事件有关的显著特征提取技术。

4.1.1 谱包络

谱包络(spectral envelope)包含着丰富的信息,可看成一个缩小的功率谱(DFT 幅值)。谱包络的形状往往由频谱中主要峰的位置所决定。格雷的一个重大发现是通过谐波幅值包络的线段近似算法(line-segment approximation)获得的(Grey 1977),他以 100 : 1 的数据衰减量很好地将原信号进行了再合成——每个谐波振幅轨迹只采用了原信号的部分信息(相当于对谐波振幅包络进行抽样)就能对原信号的音色进行再合成,合成后的信号听起来几乎与原信号一样。对于乐音的谐波分布,有学者认为,忽略基频,则超过 5 阶至 7 阶的谐波不可能独立存在。研究还显示,高次谐波的感知不是独立的,而是整体的(Howard, Angus 2001)。为进一步支持霍华德的理论,根据帕特博的研究结果(Puterbaugh 1999),有学者研究认为基频低于 400Hz 的信号,只有前 10 个谐波在听感上相互独立,高于 10 阶的谐波很大程度上只影响音色。确定谱包络的算法有很多,一种称为功率谱显著峰值检测法,另一种称为 STFT 帧的 RMS 值法,还有一种称为 DFT 帧低通滤波法,或如 3.2 节那样用 LPC 来描述共振极点结构。

4.1.2 谱质心

谱质心(spectral centroid)是描述声音明亮度(brightness)的音色特征量。这一重要参数在过去有许多研究(Helmholtz 1954, Lichte 1941),通过实验发现了该参数可基于多维尺度 MDS (multidimensional scaling)进行量度(Krumhansl 1989, McAdams *et al.* 1995, Lakatos 2000)。谱质心可看成是频谱重心所在的频率分量。常用变量包括均值、标准差、平方幅值、对数幅值、bark 频率尺度(Sekey, Hanson 1984)和谐波重心(对 bark 频率尺度的讨论详见附录)。在声音的时域历程中,谱质心不是静止而是变化的,此外,谱质心的位置也会随声音强度而变化。例如,小号在中央 C 上的弱音动态变化就会导致不同的谱质心特征。这一特征可间接用于波表合成与其他声音合成算法,这些算法经常在波形的不同位置上使用滤波器,使声音在变得更亮或更暗中调整波形,以模仿真实乐器的声音动态特性。目前,谱质心是 MPEG - 7 [动态图像专家组(Motion Picture Experts Group)]标准中描述音色的参数之一,定义如式 4.1:

$$SC_{Hz} = \frac{\sum_{k=1}^{N-1} k \cdot X^d[k]}{\sum_{k=1}^{N-1} X^d[k]} \quad (4.1)$$

其中, $X^d[k]$ 是频率帧 k 的幅值, N 是 DFT 变换长度, SC 是谱质心, 单位为 Hz。一般来说, 由谱质心可知, 声音“暗淡”则低频能量丰富, 声音“明亮”则高频能量丰富 (Backus 1976)。

也有学者认为, 谱质心可被音高或基频 f_0 归一化, 从而使谱质心成为一个无量纲的参数, 如式 4.2 (Kendall, Carterette 1996)。因此, 有研究者便关注于讨论谱质心的归一化值与绝对值的应用 (Eronen, Klapuri 2000), 以及用该参数进行音色识别 (Park 2004)。

$$SC_{\text{relative}} = \frac{\sum_{k=1}^{N-1} k X[k]}{f_0 \sum_{k=1}^{N-1} X[k]} \quad (4.2)$$

随着复合音中基频 f_0 数量的增加, 该参数只适用于如打击乐的有调声或基频缺失的非谐和音色的判别。显然, 为了获得准确的谱质心, 对音高提取算法的精度要求很高, 这也不是一个容易解决的问题。

4.1.3 波动度与抖动度

波动度 (shimmer) 与抖动度 (jitter) 分别指的是谐波的幅值与频率的短时、轻微和不规则的变化。特别是对于乐音的高次谐波, 波动度是随机振幅调制特征。抖动则是频率调制特征, 对于弦乐器组而言尤其是如此。对于弓弦乐器, 弓与弦之间的不稳定作用就会引起频率抖动——即微小的连续振动, 如拉弓和释音阶段。幅值波动度与频率抖动度的计算意味着检测谐波幅值和频率随时间的变化, 可设计一个鲁棒性较好的谐波侦测器进行检测。然而, 这种方法实现起来比较困难。一种计算抖动度与波动度的替代算法是采用 24 子带 bark 尺度。每个带对应一个谐波, 如果发现一个带存在两个或两以上的谐波, 则提取谐波频率的均值对应带的中心频率。波动度与抖动度的分布具有类噪特征, 通常认为呈高斯正态分布。

4.1.4 频谱通量

频谱通量 (spectral flux) 定义为随时间变化的帧到帧的能量波动。该参数可通过两个相邻 STFT 帧之间的能量差计算, 如式 4.3:

$$SF = |X_f^d[\cdot] - X_{f-1}^d[\cdot]| \quad (4.3)$$

$X[\cdot]$ 为帧的幅值分量, 下标 f 和 $f-1$ 分别表示当前帧与前一帧。频谱通量 SF 有时也称为谱幅值变化度 (delta magnitude spectrum)。由于语音信号的谱幅值变化速度一般比音乐信号快, 且语音具有第 3 节介绍的辅音和元音特征。而音乐信号的幅值变化剧烈程度较轻, 因此该参数已被应用于语音和音乐信号的分离。

4.1.5 对数谱分布

对数谱分布 (log spectral spread) 是由多尺度检验 (塔克 Tucker 2001) 发现的一个显著分布维度, 它定义为上下限频率之间的谱能量分布。其中上下限频率根据以频谱最大幅值设定的阈

值确定(例如,与最大幅值相差-10dB 的频点),上限和下限频率边界确定后,再对结果取对数。该分布形态与格雷发现的频谱分布(Grey,1977)存在着某些相似之处。不过该分布不是一个定量值。沿着包络变化的对数谱分布有助于通过丰满度区别音色,如长号、法国号和大号普遍缺乏高频分量,而小号则音色明亮且具有高次谐波(Howard, Angus 2001)。对数谱分布的基本算法如图 4.1,最后输出的谱分布位于上限频率(u)和下限频率(w)之间,对数谱分布也是 MPEG-7 标准的声音描述参数之一。

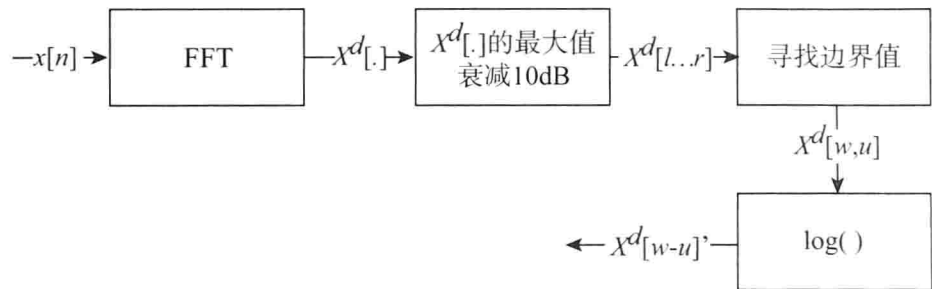


图 4.1 对数谱分布流程图

4.1.6 滚降

单位为 Hz 的滚降(roll-off)频点定义为幅值下降至功率谱总能量的 85%所对应的临界频率。该参数通常也指谱形状倾斜(skew),有时用于从稳态声(如元音)中区分打击声和高瞬态声(即高频分量丰富的声音)。滚降的定义如式 4.4 和式 4.5,其中 $X^d[k]$ 为频率帧的幅值, k 为频率帧序数, R 为能量为总能量 85%的滚降频点。

$$SR = R \tag{4.4}$$

$$\sum_{k=0}^R X^d[k] = 0.85 \sum_{k=0}^{N-1} X^d[k] \tag{4.5}$$

4.1.7 激励时间

传统意义上声音激励时间(attack time)或上升时间(rise time)定义如式 4.6,该式也可不取对数。该参数可用声音起始时刻(由阈值确定)至最大幅值时刻所经历的时间计算(见第 2 章对 ADSR 过程的介绍)。而对数上升时间是在 MDS 空间中被发现的,其他对音色的研究也发现一个既定的 MDS 音色空间往往是三维的(Saldanha, Corso 1964; Scholes 1970; Krimphoff 1993; McAdams 1995; Lakatos 2000)。

$$LRT = \log(t_{\max} - t_{\text{thresh}}) \tag{4.6}$$

式 4.6 的变化形式为式 4.7:

$$LRT = \log(t_{\text{threshMax}} - t_{\text{threshMin}}) \tag{4.7}$$

两者唯一的区别是,式 4.7 引入了一个信号最大幅值的最大阈值系数,通常设为最大幅值的 2%(Misdariis, Smith, Pressnitzer, Susini, McAdams 1998)。这种方法尽管有时对某些包络的适应较好,但这是一种固定系数算法,对于动态特征明显的信号,激励时间的计算存在较大误差。

4.1.8 颤动振幅调制(颤音)

之前我们已经介绍了振幅调制,它基本上由两列波动叠加产生。振幅调制在音乐中,特别是音乐的现场演出中无所不在,它常常与频率调制同时出现并相互强烈耦合(Backus 1976)。振幅调制频率一般为 4~8Hz,这一调制特征通常会在调制后声音的稳态部分找到。可通过估计该稳态部分的基频来计算振幅调制频率,如图 4.2。对于更高的调幅频率,如 9~40Hz,声音容易变得粗糙(roughness)且有颗粒感(graininess),也会影响最终的音色质量。

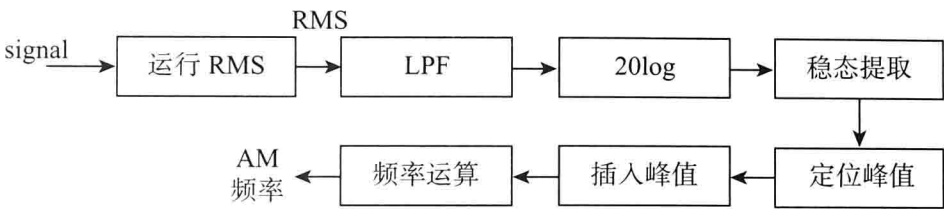


图 4.2 AM 提取框图 (Park 2000)

4.1.9 时间重心

时间重心(temporal centroid)是描述声音信号瞬态特性的重要参数,基本上是指信号时域能量重心所在的时刻。它与对数激励时间和谱质心都是 MDS 模型实验中的重要参数(Lakatos, 2000)。时间重心定义为时域信号的加权能量均值,如式 4.8,其中 $x[n]$ 为输入信号, n 为时间序号, L 为信号长度。该算法本质上与谱质心的计算方法相同——谱质心位于频域,而时间重心位于时域。时间重心也是 MPEG-7 标准用于描述音频信号的参数之一。

$$TC = \frac{\sum_{n=1}^{L-1} nx[n]}{f_0 \sum_{k=1}^{N-1} x[n]}$$

(4.8)

4.2 音乐信息检索 MIR

随着数字化革命的进行,音乐信息检索 MIR 领域的研究在信息存储、界面接口、信息检索、信息访问和管理方面都得到了迅猛发展,出现了众多音乐录音、乐谱、文章或评论,极大丰富了人们的听觉体验。不过从某种意义上说,虽然数字技术给音乐带来诸多益处,只需要按下一个按钮就可以获得人们想要的音乐,但巨大的信息量也给音乐资料的共享、管理与检索带来了噩梦。据调查,仅是 1999 年新发行的 1 万张音乐专辑就需要新注册 10 万个信息条目(Uitdenbogerd 1999)。不管人们是否意识到,随着家庭数字音频工作站、MP3(MPEG-1 音频第 3 层编码标准)播放器和具备 MP3 播放功能的移动电话或智能手机的日益普及,数字音乐播放已经成为了一种通用的必要配置,因此就不难想象音乐所带来的数字信息的巨大膨胀!自 20 世纪 90 年代以来,音乐信息检索量的增长也导致了研究者对 MIR 领域关注度的增加。目前为止,已经取得了很多应用研究成果,如哼唱检索,节拍检测,乐器识别,流派分类,艺术家识别,智能音乐分析,乐谱识别等。其中显著特征提取算法在绝大多数应用中都起到了极其关键的作用,本节将要讨论的专题也与之前提到的该算法有关。

4.2.1 哼唱检索 QbH

哼唱检索(Query-by-humming)是一种旋律自动识别技术,它通过向机器(如计算机)输入一段哼唱旋律,将输入信号与既有人工系统数据库进行最接近匹配查询后,输出一份可能匹配项

的排名列表。哼唱检索系统的典型设计方法是声音信号(歌手的哼唱旋律)转换层附近自下而上的模型结构,并提取这些哼唱信号的音高和节奏信息(如激励时间、时域参数和片断等)。接下来通过将这些特征参数输入至旋律分析模块,在该模块中测量哼唱信号与既有数据库旋律的相似度。早期的 MIR 系统通常采用 MIDI 文件(乐器数字接口协议)进行旋律相似度检测(Kurth 2002;Pardo 2004;Ghies 1995)。MIDI 文件的优点在于内嵌了准确的旋律(如果输入正确的话)、音高、节奏和时值以及力度(相当于动态)等参数的数字化信息。在哼唱检索系统中,由于数据库本身是单声道,则查询输入信号也应该是单声道(即单个哼唱声)。节奏信息是哼唱检索中一个重要的参数,也是检索旋律的核心参数,如有研究论文表明,仅用该节奏信息就可以确定哼唱信号的目标旋律(Jang *et al.* 2001)。然而,时域与频域结合分析似乎能得到更好的匹配结果。因此,一旦提取了含有音高信息的片断,则该特征向量可用于各种模式识别模型,如隐含马尔可夫模型(Jang 2005),高斯混合模型(Marolt 2004)或适应性增强模型(Park 2005)。此外哼唱检索的主要问题不仅是设计一个准确的音高提取算法和节奏分析器,还需要设计一个足够灵活的纠错系统,用于修正歌手哼唱输入中如音准或节奏上的微小错误(Kurth 2002)。

4.2.2 自动节拍检测与节奏分析

音乐检索另一个有趣的研究领域是节拍检测(beat detection)和节奏分析(rhythm analysis)。节拍、节奏和节奏型(tempo)都是音乐中非常重要的参数,并经常用于分析音乐的片断、织体和结构,特别是从时域而不是频域的角度来分析曲式时显得尤为重要。有趣的是,虽然乐谱包含了节奏和音符的时值信息,但同一乐谱的实际演奏却因人而异、因乐队而异(有时差异很大),甚至即使大家阅读的是同一份乐谱,对时值的解释和演奏出的音乐也会在多个层面上完全不同,特别是对时间标记的把握——节奏型、节奏与节拍。此外,就算演奏该曲目的乐队只有两人,那么听众对节奏、节奏型与韵律的感知实际上也会有轻微差别——这种差异往往取决于事先对曲目背景知识的了解程度(McKinney 2004)。由于人为因素导致个体差异和音乐体验的不同,因此设计一个具有良好鲁棒性的节拍、节奏和速度检测系统是非常困难的。目前用来设计该系统的技术包括:针对钢琴曲的基于相位和周期锁定的单振荡发生器(Pardo 2004),利用频域和谱通量联合分析进行的周期检测和节拍时域位置估计(Alonso 2004),运用高斯模型与相关性分析进一步从格式塔心理学理论构建的节奏解释模型(Frieler 2004),用 k 均值(k-means)距离等算法进行的节拍测量(Takeda 2004),用欧氏区间向量距离进行的节拍测量(Coyle 1998),用汉明距离进行的节拍测量(Toussaint 2004),以及用音程差距离(interval-difference distance, Coyle, Shmulevich 1998)和所谓交换距离(Orpen 1992)进行的检测。

4.2.3 自动音色识别

音乐音色的概念与感知一直以来都是模糊和神秘的,事实上它的定义无论在心理学、音乐学或计算机科学中都没有进行清晰的界定。尤其是在音色的早期研究中,都是采用如冷、温、凉、无光泽、短、长、光滑、粗糙等定性词语进行描述,而这些形容词的语义都很难确切地描述音色的结构、维度或物理机制特征。人们可能会有疑问:“人类识别和区分音乐音色很容易,那么为什么设计一台音色检测器如此困难呢?”其实在许多情况下,用计算机系统来模拟人类的嗅觉、触觉或视觉,并准确地识别所感觉到的对象是非常困难的。似乎在许多情况下,对于感知一类的任务,机器和人类的完成能力呈反比关系,或者说机器容易完成的任务,人类可能需要花很

大的精力才能完成,反之亦然。不过持续的研究表明,尽管人类对音色的感知相当复杂,但研究者已经逐渐对决定音色复杂性的因素有了相当的认识。那么,是什么使音色如此难以解释呢?答案当然过于复杂,不过分析起来可能有以下几个主要原因:

- ① 音色是声音感知的主观属性,而不是一个纯粹的物理属性,从某种程度上说它是一个无形的实体。
- ② 音色是一个多维属性,需要充分了解影响音色的特征参数的数量和各自的重要性才能理解音色的真正含义(已知可用于量度音色的维度参见 4.1 节)。
- ③ 目前没有发现合适的判断音色的主观尺度。
- ④ 目前没有统一的音乐信号标准集,供研究者用于测试已开发的音色计算模型。

目前的乐器音色机器识别模型一般可分为两大类,自上而下(top-down)的模型和自下而上(bottom-up)的模型。大多数音色识别研究使用自下而上的模型,也称为数据挖掘模型(data-driven model),如图 4.3。计算机系统通常采用监督学习(supervised learning)和非监督学习(unsupervised learning)技术来实现智能识别。目前乐器音色识别的初步研究所采用的方法包括 k 邻近值法(Fujinaga 2000),贝叶斯分类器法(Martin, Kim 1998),二叉树法(Jensen, Arnspang 1999),基于径向椭圆基函数的人工神经网络(Park 2004),独立子空间分析法(Vincent 2004),高斯混合模型(Essid 2004)以及支持向量机(Marques 1999)。至于实际 MIR 系统的音色识别能力取决于 GIGO 冗余量进出模式(garbage-in-garbage-out)的判决性能,该性能取决于系统的鲁棒性特征。

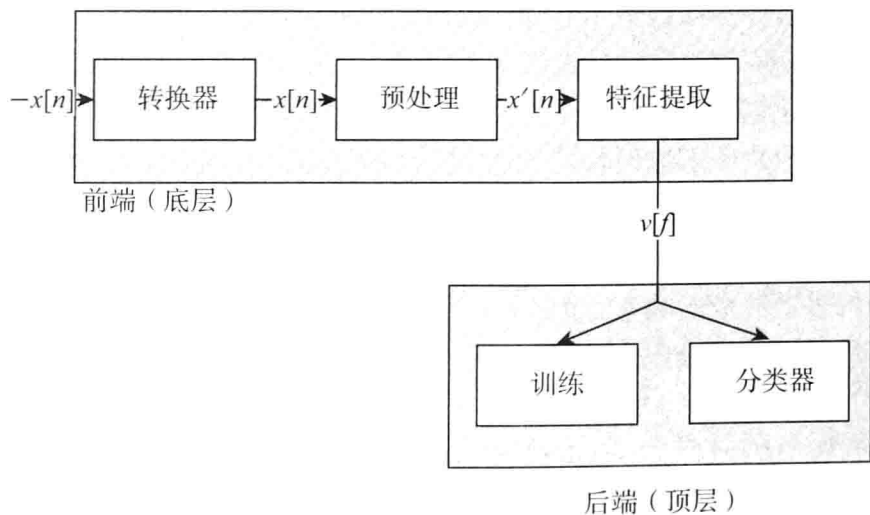


图 4.3 音色分类的 Bottom-up 模型

4.3 特征调制合成系统 FMS

特征调制合成系统 FMS(feature modulation synthesis)的研究动机来自于可供作曲家们缺乏用于合成与再合成的工具,他们希望直接处理与声音感知有关的参数,同时能够用特征调制的方式控制和修改这些参数(Park 2004; Park et al. 2007)。经典的合成算法往往使用正弦波或抽象的噪声通过参数提取进行合成,而这些参数似乎与声音的实际感知没有太多联系。FMS 系统试图缓解这种控制参数的提取与用特征参数调制控制音色相脱节的情况。该系统的结构也是基于分析-合成模型,包括显著特征提取模块、特征调制模块与合成模块,如图 4.4,其中 x 为输入的声音事件, x' 为经调制合成后的结果, v 和 v' 则分别是分析模块所提取的和经调制后的特征

向量。在 FMS 系统中合成声音的过程是:首先分析输入声音事件的显著特征,然后选择一组特征向量进行调制,最后再合成新的声音事件,同时保持所有与音色有关的特征量不变。虽然对音色的某一个维度进行调制在理想情况下不会影响其他维度的参数,但实际上会产生一些人为失真。图 4.4 的第一个模块(显著特征提取模块)利用了各种时域和频域的分析算法来提取特征参数和特征向量(Park *et al.* 2007, 2008);第二个模块为特征调制模块,它按所需的效果修正各特征参数,是 FMS 系统的重要模块,该模块最终将特征向量 v 修改为所需的特征向量 v' ;最后一个模块是通过时域和频域算法的结合,将调制后的特征向量 v' 合成为新的声音。FMS 系统的调制类型包括谱质心调制、RMS 幅值包络调制、谱分布替换、谐波扩展和压缩、基于 LPC 的残差噪声调制以及三维谱包络调制替换。

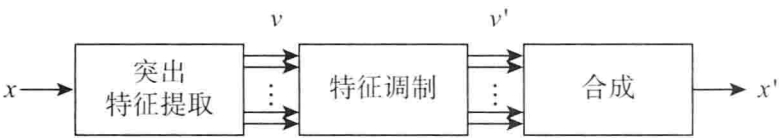


图 4.4 基本 FMS 结构

为了更好地了解 FMS 系统的工作过程,接下来考虑这样一个例子:改变电贝斯声音的明亮度。通常情况下,吉他的弦音在波形的激励阶段听起来最为明亮,然后能量随着时间的衰减使音色变得暗淡——正如第 7 章的物理合成模型实例所述,决定声音明亮度的高频分量的衰减率较快。通过分析信号的谱质心可知,波形的明亮度是时间的函数,因此,每一个频率帧乘以适当的谱包络可以改变谱质心从而改变信号的明亮度。谱质心的调节方法就像一个拉锯模型,如果希望低频能量比更高则应使锯向 DC 直流分量倾斜,反之希望高频分量具有更高的能量,则向奈奎斯特频率倾斜。拉锯模型的初始平衡点位于“奈奎斯特频率/2”处,然后向 DC 直流分量还是奈奎斯特频率倾斜取决于调制时所需要的谱质心的位置,如图 4.5 所示,拉锯的形状形成了用于将谱质心向 DC 直流分量或奈奎斯特频率平移的乘积包络。FMS 系统的另一个应用实例是,可通过基于 LPC 算法的声音残差信号计算(详见 3.2 节)进行声信号的噪声分析。而理想的噪声残差信号可表示波形的噪声,同时还可通过对噪声残差信号进行调制,并利用信源滤波器模型来控制既定信号的信噪比。

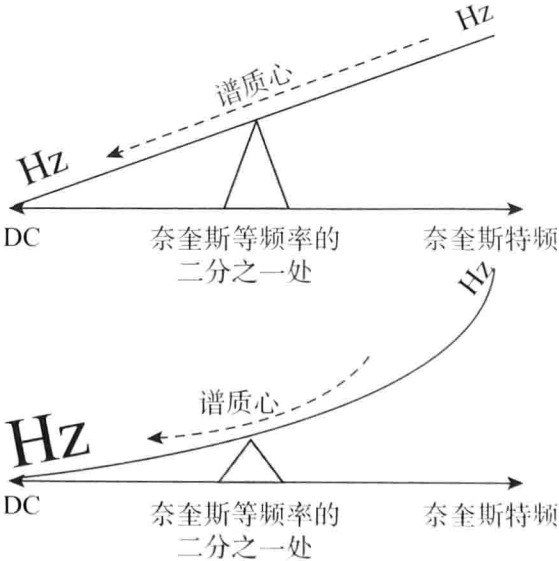


图 4.5 谱质心调制的拉锯模型

5. 乐例分析

声码器之所以在音乐中的应用如此普及——不仅仅因为它为音乐家的创作提供了更多更灵活的可能性,也许还因为语音本身就是人类最为熟悉的声音。许多作曲家和艺术家都使用了声码器来创作他们的作品,从克鲁夫威克(Kraftwerk)的《死去的机器人》(*Die Roboter*, 1978)到劳丽·安德森(Laurie Anderson)的《哦,超人》(*O Superman*, 1981),从赫比·汉考克(Herbie Hancock)的《摇滚》(*Rock It*, 1983)到歇尔(Cher)的《相信》(*Believe*, 1999),再从达夫特·朋克(Daft Punk)的《又一次》(*One More Time*, 2000)到史努普·多戈(Snoop Dog)的《感官诱惑》(*Sensual Seduction*, 2007)等。在计算机音乐创作与分析中应用声码器的先驱包括查尔斯·道奇(Charles Dodge)和保罗兰·斯基(Paul Lansky)。查尔斯·道奇于20世纪60年代开始在贝尔实验室中对LPC进行研究,并于1972年发表了音乐作品《演讲歌曲》(*Speech Songs*)。在贝尔实验室工作时,道奇通常在其他研究人员不在的晚上采用最先进的语音技术进行音乐创作。《一个人坐在咖啡厅里》(*A Man Sitting in the Cafeteria*)是四首《演讲歌曲》(*Speech Songs*)作品的典型范例。人们可以在该作品中清楚地听到语音和非语音成分是如何混合在一起的,语音因此具有了音乐性,尽管这已经不是传统意义上的语音和音乐了。无可否认,经作曲家进行节奏和音高调制后合成的声音也许变得愉快和幽默,甚至能够产生其他一些令人意想不到的听觉和审美效果。该作品的原文本内容如下:

*"A man sitting in the cafeteria
had one enormous ear
and one tiny one
Which was fake?"*

保罗·兰斯基对于LPC算法在音乐中的应用也进行了大量研究,并发表了众多音乐作品,其中最著名的是《絮叨》(*Idle Chatter*)系列作品。第一首《絮叨》作品发表于1985年,该作品将听众带入了声音诗歌(sound poetry)和演讲世界之旅。作曲家使用人们熟悉的语音信号进行处理,掩盖了其多维结构信息,以让听众感觉到这些说话声但却无法分辨说的是什么。单纯从音乐性或声音效果的层面而不从语言的层面上看,这首曲子可看作是对快节奏声音序列与模式以及吞音经验的模仿。从另一个层面上看,该作品将语音材料进行打击乐化与旋律化处理,模糊了规范语音音色与方言语音音色的界限。与以上作品呈现方式相反的例子是笔者创作的《母亲》(*Omoni*, 2000)和《父亲》(*Aboji*, 2001),这两首作品采用了语言叙述的方式来处理语音素材。在诸如此类的作品中,语音材料往往抽象而不完整,并以各种方式进行调制,而具体的材料则包括文字、诗歌、复杂形态的语言和音乐素材本身等。在某种程度上,至少从笔者的角度看,创作者试图在平衡自己作为作曲家和记者的角色,从而让作品具有音乐性的同时又具有纪录性和真实性。在本书结束之前,将要介绍的最后一个作品来自于安迪·穆尔(Andy Moor)的《狮子的成长》(*Lions are Growing*, 1978),该作品基于理查德·布劳提根的诗歌《隆美尔继续驱车到埃及》(*Rommel Drives on into Egypt*)创作,被收录于一张名为《数字域的诠释》(*The Digital Domain: A Demonstration*, 1983)的CD唱片中。与保罗·兰斯基和查尔斯·道奇的作品应用LPC技术不同的是,穆尔利用了相位语音合成技术来变换声音的纹理、共振峰结构、音高和时域特性等特征。在穆尔的作品中,可能会听到

一些类似是由查尔斯·道奇的作品《演讲歌曲》片断进行合成处理的素材,六年后使用最新的语音合成算法处理了同样的素材——这也许是一个巧合(但至少语音声码器已经成为了音乐创作的重要角色)。就想法本身来说,与道奇的作品有些类似,但在声音质量上明显上了一个台阶,并且它还运用了多种合成方法。该作品最吸引人、也是最幽默和令人感到愉快的地方是(也有许多由滑音产生的移时效果),结尾的一句——猫王式的“*thank you, thank you (very much)*”。不过在本书行将结束之际,笔者同样也愿意真诚地向读者说一声:谢谢,非常感谢您的阅读与支持!

参考文献与扩展阅读

- [1] Alonso, M., B. David and G. Richard 2004. “Tempo and Beat Estimation of Musical Signals,” in ISMIR 2004 Fifth International Conference on Music Information Retrieval Proceedings, pp. 158 – 163.
- [2] Backus, J. 1976. “Input Impedance Curves for the Brass Instruments”. Journal of the Acoustical Society of America, 60, pp. 470 – 480.
- [3] Bisnhu, Atal, M. R. Schroeder 1967. “Predictive coding of speech,” in Proc. Conf. Communications and Proc., Nov. 1967, pp. 360 – 361.
- [4] Bisnhu, Atal, M. R. Schroeder 1970. “Adaptive predictive coding of speech,” Bell Syst. Tech. J., Vol. 49 No. 8, pp. 1973 – 1986, Oct. 1970.
- [5] Bogert, B. P., M. Healy and J. Tukey 1963. “The quefrency alalysis of time series for echoes: Cepstrum, pseudo-autocovariance, cross-cepstrum, and saphe cracking,” in Time Series Analysis, M. Rosenblatt (Ed.) 1963, Ch. 15, pp. 209 – 243.
- [6] Coyle, E. J. and I. Shmulevich 1998. “A system for machine recognition of music patterns”. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Seattle, Washington.
- [7] Childers, D., D. Skinner and R. Kemerait 1977. “The Cepstrum: A Guide to Processing”, Proceedings of the IEEE, Vol. 65, No. 10, pp. 1428 – 1443.
- [8] Cuadra, P., A. Master and C. Sapp 2001. “Efficient Pitch Detection Techniques for Interactive Music”, Proceedings of the 2001 ICMC, Havana.
- [9] Dolson, M. 1986. “The phase vocoder: a tutorial,” Computer Music Journal, Spring, Vol. 10, No., 14 – 27.
- [10] Doyle, M. “Private communication.” October 13, 2000. The extent of actual use by Roosevelt and Churchill is still unclear and needs further research to clarify; however, there is evidence of a call between Truman and Churchill on VE-Day (Victory in Europe).
- [11] Dudley, H. 1938. “System for the Artificial Production of Vocal and Other Sounds (Voder),” U.S. patent 2,121,142.
- [12] Dudley, H. 1939. “The Vocoder”, Bell Labs. Rec., Vol. 17, pp. 122 – 126.
- [13] Dudley, H. 1939. “The Vocoder”, Journal of Acoustical. Society of America, Vol. 11, No. 2, p. 169.
- [14] Eronen and A. Klapuri 2000. “Musical Instrument Recognition Using Cepstral Coefficients

and Temporal Features”. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2000, pp. 753 – 756.

[15] Essid, S., G. Richard and B. David 2004. “Musical instrument recognition based on class pairwise feature selection” Proceedings of ISIMIR 2004, Barcelona, Spain.

[16] Flanagan, J. and R. Golden 1966. “Phase vocoder”. Bell System Technical Journal, 45, 1493 – 1509.

[17] Flanagan, J. 1972. “Voices of men and machines”, Journal of the Acoustical Society of America, Vol. 51, pp. 1375 – 1387.

[18] Frieler K. 2004. “Beat and meter extraction using Gaussified onsets”. Proceedings of ISIMIR 2004, Barcelona, Spain.

[19] Gihis A. 1995. “Query By Humming — Musical Information Retrieval in an Audio Database”. ACM Multimedia 95, Electronic Proceedings November 5 – 9, 1995 San Francisco, California.

[20] Grey, J. M. 1977. “Multidimensional Perceptual Scaling of Musical Timbre”. Journal of the Acoustical Society of America Vol. 61, pp. 1270 – 1277.

[21] Helmholtz, H. L. 1954. On the Sensation of Tone as a Physiological Basis for the Theory of Music (translation of original text 1877), New York; Dover Publications.

[22] Herrera, P., X. Amatriain, E. Batlle and X. Serra 2000. “Towards instrument segmentation for music content description; a critical review of instrument classification techniques”, in International Symposium on Music Information Retrieval, pp. 23–25, Plymouth, Mass, USA.

[23] Howard, D. M. and J. Angus, 2001. Acoustics and Psychoacoustics. Oxford, Boston: Focal Press.

[24] Jensen, K. and J. Arnspang 1999. “Binary Decision Tree Classification of Musical Sounds”. In Proceedings of the 1999 International Computer Music Conference.

[25] Jang, J., C. Hsu and C. Yeh 2001. “Query by Tapping: A New Paradigm for Content-Based Music Retrieval from Acoustic Input”, Proceedings of the Second IEEE Pacific Rim Conference on Multimedia: Advances in Multimedia Information Processing, pp. 590 – 597, October 24 – 26.

[26] Jang, J., C. Hsu and H. Lee 2005. “Continuous HMM and Its Enhancement for Singing/Humming Query Retrieval”. Proceedings of the ISMIR 2005 Conference, pp. 546 – 551.

[27] Kondoz, Ahmed 2004. Digital Speech; Coding for Low Bit Rate Communication Systems, Wiley.

[28] Krimphoff, J., S. McAdams and S. Winsberg 1994. “Characterisation du Timbre des Sons Complexes. II Analyses acoustiques et quantification psychophysique”. Journal de Physique IV, Vol. 4, pp. 625 – 628.

[29] Kurth, F. 2002. “Efficient Fault Tolerant Search Techniques for Full – text Audio Retrieval”. Proc. 112th AES Convention, Munich, Germany.

[30] Lakatos, S. 2000. “A Common Perceptual Space for Harmonic and Percussive Timbres”. Perception & Psychophysics 62(2), pp. 1426 – 1439.

[31] Lichte, W. H. 1941. “Attributes of Complex Tones”. Journal of Experimental Psychology,

28, pp. 455 – 480.

[32] Marolt, M. 2004. “Gaussian Mixture Models for Extraction of Melodic Lines From Audio Recordings”, Proceedings of ISIMIR 2004, Barcelona, Spain.

[33] Martin, K. and Y. Kim 1998. “Musical Instrument Identification: A Pattern-Recognition Approach”, 136th meeting of the JASA.

[34] McAdams, S., S. Winsberg, S. Donnadieu, G. De Soete and J. Krimphoff, 1995. “Perceptual Scaling of Synthesized Musical Timbres: Common Dimensions, Specificities, and Latent Subject Classes”, *Psychological Research* 58, 177 – 192.

[35] McKinney, M. 2004. “Extracting the perceptual tempo from music”, Proceedings of ISIMIR 2004, Barcelona, Spain.

[36] Misdariis, N., B. K. Smith, D. Pressnitzer, P. Susini and S. McAdams 1998. “Validation of a Multidimensional Distance Model for Perceptual Dissimilarities among Musical Timbres”, 16th International Congress on Acoustics and 135th Meeting Acoustical Society of America, Seattle, Washington.

[37] Moorer, J. A. 1975. “On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer”, Doctoral Dissertation, Department of Computer Science, Stanford University.

[38] Kendall, R. A. and E. C. Carterette 1996. “Difference Threshold for Timbre Related to Spectral Centroid”, Proceedings of the 4th International Conference on Music Perception and Cognition, pp. 91 – 95.

[39] Orpen K. and D. Huron 1992. “Measurement of similarity in music: A quantitative approach for nonparametric Representations”, In *Computers in Music Research*, Vol. 4, pp. 1 – 44.

[40] Pardo, B. 2004. “Tempo Tracking with a Single Oscillator”. Proceedings of ISIMIR 2004, Barcelona, Spain.

[41] Park, T. H. 2000. “Salient Feature Extraction of Musical Signals”, Master’s Thesis. Dartmouth College, Electro-acoustic Music Program.

[42] Park, T. H. 2004. “Towards Automatic Musical Instrument Timbre Recognition”, Ph.D. dissertation, Princeton University.

[43] Park, T. H., Z. Li and J. Biguenet 2008. “Not Just More FMS: Taking it to the Next Level”, In Proceedings of the 2008 ICMC Conference, Belfast, Ireland.

[44] Parker, C. 2005. “Applications of Binary Classification and Adaptive Boosting to the Query-by-Humming Problem”. Proceedings of the ISMIR 2005 Conference.

[45] Puterbaugh, J. D. 1999. “Between a Place and Some Location, A View of Timbre through Auditory Models and Sonopoietic Space”, Ph.D. Dissertation. Princeton University Music Department.

[46] Rabiner, L. R. and R. W. Schafer 1978. “Digital Signal Processing of Speech Signals”, Prentice Hall, Signal Processing Series.

[47] Scholes, P. A. 1970. *The Oxford Companion to Music*. London: Oxford University Press.

[48] Schroeder, M. R. 1968. “Period Histogram and Product Spectrum: New Methods for Fundamental Frequency Measurements”, *Journal of Acoustical Society of America*, Vol. 43, No. 4, pp. 829 – 834.

- [49] Sekey, A. and B. A. Hanson, 1984. "Improved 1-bark Bandwidth Auditory Filter", JASA Vol. 75, No. 6.
- [50] Takeda, H. 2004. "Rhythm and Tempo Recognition of Music Performance from a Probabilistic Approach", Proceedings of ISIMIR 2004, Barcelona, Spain.
- [51] Toussaint, G. 2004. "A Comparison of Rhythmic Similarity Measures", Proceedings of ISIMIR 2004, Barcelona, Spain.
- [52] Truax, Barry 1978. Handbook for Acoustic Ecology. A.R.C. Publications, Vancouver.
- [53] Tucker, S. 2001. "Auditory Analysis of Sonar Signals", Ph. D. Transfer Report. Department of Computer Science. University of Sheffield.
- [54] Uitdenbogerd, A. and J. Zobel, 1999. "Matching techniques for large music databases", Proceedings of the 7th ACM International Multimedia Conference, pp. 57 – 66.
- [55] Vincent, E. and X. Rodet 2004. "Instrument identification in solo and ensemble music using Independent Subspace Analysis", Proceedings of ISIMIR 2004, Barcelona, Spain.
- [56] Zölner, Udo 2002. DAFX — Digital Audio Effects. Editor, John Wiley and Sons.
- [57] Zwicker, E. and H. Fastl 1999. Psychoacoustics Facts and Models. Springer Verlag.

附 录

1. 伸缩还是不伸缩,是一个问题

如本书第 1 章所述,人耳对频率和音调的感知通常是非线性的。现行用于量度频率及其感知的非线性尺度标准有很多。接下来将介绍一些具有代表性的尺度标准。

1.1 等程尺度

音乐学中最常见的频率尺度也许是十二平均律。现在,它是一个被大量音乐使用的音调系统,特别是在西方音乐世界中,该律制有利于不改变音之间的相对关系进行转调。简而言之,音符之间的距离(音程)已经被结构化,由式 1.1 可知,相邻两律的音符对形成相等的频率比:

$$f_x = f_r \cdot 2^{n/12} \quad (1.1)$$

本质上,由式 1.1 可知,一个八度被分成十二“等份”,整数 n 决定了音符及其八度音的频率(音高)。设 n 取值由 $n=0$ 增大至 12,则参考音高 f_r 分别乘以 $1, 2^{1/12}, \dots, 2^{11/12}$ 和 2 而得到一个十二平均律的半音阶,最后一个乘积因子指高一个八度。两个半音的比(相邻 n 的整数值)等于一个音分,100 个音分组成一个半音音程,如式 1.2:

$$\text{音分} = 2^{1/12} \quad (1.2)$$

常用的参考频率是 440Hz,对应着 A4 音,或标准钢琴上音高上升方向的第 49 个键所对应的音高。因此,寻找或计算等程律中的音符非常直观。例如,因为 C4 音比 A4 音低九个“音符”,则中央 C4 音(261Hz)可由式 1.3 计算:

$$\begin{aligned} f_x &= f_r \cdot 2^{n/12} \\ &= 440 \cdot 2^{\frac{-9}{12}} \\ &= 261.6\text{Hz} \end{aligned} \quad (1.3)$$

1.2 调音

调音是指基于理想谐波结构的音阶校准系统,即音符包含的频率分量呈整数比(有理数)。与等程律不同,调音是将乐器的音高调至你所需的音阶上——如对于钢琴或带品弦乐器等音高固定的乐器,A 大调(A Major)的设置就不适用于 B 大调(B Major)的设置,转调会有问题。因此,尽管等程律现在已经在音乐或非音乐领域得到了广泛应用(至少在流行音乐领域,很少听到其他不符合等程律系统的调音标准),但从某种意义上说,调音系统比等程律系统更为精确,因为它是基于谐波频率结构的系统。

1.3 Bark 尺度

Bark 频率尺度(bark frequency scale)是一种非线性频率尺度,它将以 Hz 为单位的频率映射

到以心理声学为基础的 24 个子带上,每个子带的单位称为 bark。该尺度更接近人耳的非线性听觉系统,一个 Bark 对应一个临界带(critical band)。24 个临界带所对应的中心频率及边界频率如表 1.1 所示。有大量公式用于计算 Bark 尺度,其中一个如式 1.4(Zwicker, Terhardt 1980):

$$B = 13 \tan^{-1} \left(\frac{0.76f}{1000} \right) + 3.5 \tan^{-1} \left(\frac{f}{7500} \right)^2$$

(1.4)

表 1.1 Bark 频率尺度及其中心频率与带宽边界频率

Bark 单元	Bark 中心频率	带宽边界频率
1	50	0, 100
2	150	100, 200
3	250	200, 300
4	350	300, 400
5	450	400, 510
6	570	510, 630
7	700	630, 770
8	840	770, 920
9	1000	920, 1080
10	1170	1080, 1270
11	1370	1270, 1480
12	1600	1480, 1720
13	1850	1720, 2000
14	2150	2000, 2320
15	2500	2320, 2700
16	2900	2700, 3150
17	3400	3150, 3700
18	4000	3700, 4400
19	4800	4400, 5300
20	5800	5300, 6400
21	7000	6400, 7700
22	8500	7700, 9500
23	10500	9500, 12000
24	13500	12000, 15500

1.4 Mel 尺度

和 Bark 一样, Mel 频率尺度也是一种频率映射感知模型,它反映的是对等距频率的感知。

Mel 尺度描述了音高感知的非线性映射,它作为频率的函数如式 1.5 (Stevens, Volkman 1937, 1940):

$$m = 1127.01048 \log_e \left(1 + \frac{f}{700} \right)$$

(1.5)

由式 1.5 可知, $f=1\text{kHz}$ 时, f 等于 m ——即 1kHz 是 mel 频率与以赫兹为单位的真实频率之间的参考点, $1\text{kHz}=1000\text{ mel}$ (参考分贝值为高于人耳听觉阈限的 40dB)。Mel 一词来源于音乐术语“melody”, 是旋律的频率分量及其与感知有关的音高之间的距离的量度。

对式 1.5 两边进行自然指数运算, 可将 mel 尺度转化为频率 (Hz) 尺度, 得式 1.6:

$$f = 700 \cdot e^{\frac{m}{1127.01048}} - 1$$

(1.6)

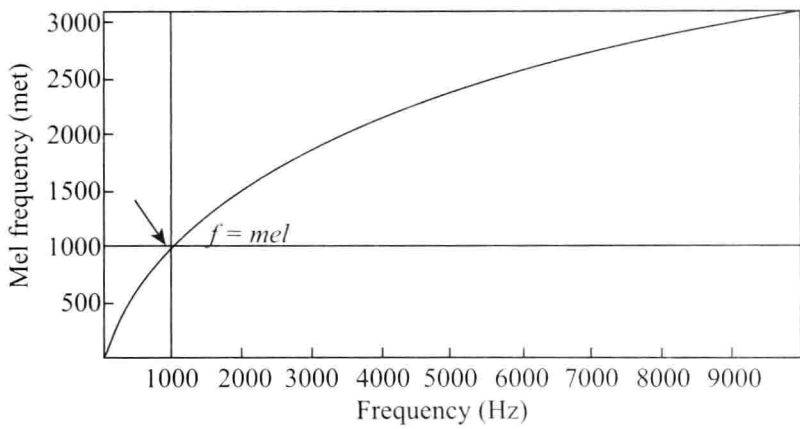


图 1.1 $f=0\sim10\text{kHz}$ 的 Mel 频率与 Hz 频率关系图

2. 本书的 MATLAB® 程序

本节的内容包括了本书大多数图像、算法和实例所使用的 MATLAB® 的程序。希望通过实践和尝试, 这些源代码对读者的学习和进一步研究有所帮助。这些程序可从网址 <http://music.princeton.edu/~park/dsp-book> 中下载。

表 2.1 本书的 MATLAB® 源程序索引

章	节	程序描述	.m 程序文件名
1	2	正弦波, 图 2.2	sineWave.m
	2	生成正弦波, 代码 2.1	sineWaveSound.m
	4	采样, 图 4.1~4.4	sineWave.m
	4	采样与频率混叠, 图 4.5~4.11	samplingNAliasing.m makeSine.m
	5	幅值量化, 图 5.2	quantizationPlot.m

续表

章	节	程序描述	.m 程序文件名
	6	DC 直流信号,图 6.1	dcPlot.m
	6	DC 偏移分量的消除,代码 6.1	dcOffsetRemoval.m
	7	失真与削波,图 7.1	distortionPlot.m
	7	量化与抖动,图 7.2	ditherPlot.m
	7	谐波失真,图 7.3	ditherHarmonicDistortion1.m
	7	谐波失真,图 7.4	ditherHarmonicDistortion2.m
2	2	波形与包络,图 2.2、2.3	rmsPlots.m, rms.m
	2	反向重放,代码 2.1	reverseWave.m
	3	波形显示,图 3.1	waveformSubplots.m
	3	循环,图 3.2	waveformLoop.m
	4	时窗,图 4.1、4.3	windowingPlots.m
	5	过零率,图 5.1	zcrPlot.m
	5	过零率,图 5.2、5.4	zcrDemo.m
	6	过采样,图 6.1	upSampling.m
	6	减采样,图 6.2	downSampling.m
	6	减 / 过采样,代码 6.1	upDownSampling.m
	7	重叠 (OLA),图 7.1、7.2	olaPlots.m
3	2	粒子与时窗,图 2.1、2.3	granularWindows.m granularSynthesis.m
	3	动态压缩,代码 3.1	compressor.m
	5	等响平移,图 4.1	panning.m
4	4	节拍,图 4.1	beating.m

续表

章	节	程序描述	.m 程序文件名
	4	振幅调制,图 4.4	amplitudeModulation.m
	5	贝塞尔函数,图 5.2	bessel.m
5	4	脉冲函数,图 4.1	impulsePlot.m
6	2	频率响应类型,图 2.4	freqLinearLogBode.m
	2	低通滤波器(LPF)的频率响应,图 2.5	freqIirLpf.m
	3	幅值/相位响应,图 3.1	freqFirLpf.m
	3	幅值/相位响应,图 3.3	phaseResponse.m
	3	相位失真,图 3.5	linearVsNonlinearPhase.m
	3	群延时,图 3.6	groupDelay.m
7	2	滤波器特征, 图 2.5~2.14、图 2.16~2.27	filterDesign.m
	3	减法合成,代码 3.1	subtractiveSynthesis.m
	3	双二次带通滤波器(BPF), 代码 3.2,图 3.3、3.4	biquadBpf.m
	3	梳状滤波器系数,图 3.7	combFilterCoeff.m
	3	梳状滤波器的 z 平面, 图 3.8~3.10	combFilterZplane.m
	3	梳状滤波器 1,代码 3.3	combFilterExample1.m
	3	梳状滤波器 2,代码 3.4	combFilterExample2.m
	3	梳状滤波器 3,代码 3.5	combFilterExample3.m
	3	弹拨乐器模型,代码 3.6, 图 3.22、3.23	pluckedStringKS0.m
	3	弹拨乐器模型的直接应用, 代码 3.7	pluckedStringKS1.m
	3	合声效果器,代码 3.8	chorus.m
	3	全通滤波器的 z 平面,图 3.33	allpassFreq.m

续表

章	节	程序描述	.m 程序文件名
	3	全通脉冲响应,图 3.38	allpassImpulse.m
	4	全通脉冲响应,图 3.39~3.41	allpassImpulses.m
	4	全通混响器,代码 3.10	allpassReverb.m
8	2	加法合成,图 2.1~2.3、4.2	squareWave.m
	4	傅立叶变换基础, 图 4.2、4.4、4.6~4.8	fourierBasics.m
	6	零填充,图 6.1、6.2	zeroPadding.m
	7	频率混叠,图 7.2~7.4	samplingNaliasing2.m
	8	过采样,图 8.6	upsamplingFreqDomain.m
	9	矩形窗,图 9.1~9.7	windowingDftRecSinusoids.m
	9	时窗类型,图 9.8~9.15	windowTypesDft.m
	9	sinc 函数,图 9.16~9.19	sincPlots.m
	11	循环卷积,图 11.1~11.3	convolutionPlots.m
	12	抖动,图 12.1、12.2	ditherHarmonicDistortion2.m
	13	光谱图,图 13.1	specgramPlot.m
9	15	MATLAB® 与快速傅立叶变换(FFT),代码 14.1	fftExample.m
	2	长时平均频谱(LTAS), 图 2.2	ltas.m
	2	对数与线性 FFT,图 2.4、2.5	fftLogVsLinear.m
	2	峰谷值分析,图 2.9	peakValleyAnalysisMain.m
	2	峰谷的均方根值,图 2.8	peakValleyAnalysis.m
	2	突出峰值检测,图 2.9	peakValleyRmsMain.m peakValleyAnalysis.m
	2	反梳状滤波,图 2.11、2.12	inverseCombFilterPlots.m

续表

章	节	程序描述	.m 程序文件名
	2	反梳状滤波器提取音高,图 2.13	inverseCombFilterPitch.m
	2	倒谱音高分析,图 2.15	cepstrum.m
	3	包络跟随器,图 3.2	envFollower.m
	3	Bark 尺度频带,图 3.3	channelVocoder.m
	3	包络跟随器结果,图 3.4	channelVocoder.m
	3	语言/非语言成分的过零率分析(ZCR),图 3.8~3.10	zcrAnalysis.m zcrComputation.m
	3	语言/非语言成分的预权重能量率分析,图 3.11	preEmphasisVuv.m
	3	语言/非语言成分的低频能量率分析,图 3.12	lowFullEnergyRatioVuv.m
	3	语言/非语言成分的频谱平坦量分析(SFM)	sfmVuv.m, sfm.m
	3	线性预测编码(LPC),图 3.17	lpcAnalysis.m
	3	相位展开,图 3.22	phaseUnwrapping.m
附录	1	Mel 频率与赫兹,图 1.1	melVsHzPlot.m
		原著封面	bookCover.m

参考文献与扩展阅读

[1] Stevens, S. S., Volkman, J. and Newman, E. B. 1937. "A Scale for the Measurement of the Psychological Magnitude of Pitch", Journal of the Acoustical Society of America, 8.

[2] Stevens, S. S., Volkman, J. and Newman, E. B. 1940. "The Relation of Pitch to Frequency, A Revisited Scale", Journal of the Acoustical Society of America, 53.

[3] Zwicker, E. and Terhardt E. 1980. "Analytical Expressions for Critical-band Rate and Critical Bandwidth as a Function of Frequency", Journal of the Acoustical Society of America, 68, 1523 -1525.

索引

A

阿尔文·卢西尔 79, 181
安迪·穆尔 273, 288
安吉尔·巴德拉莫蒂 80

B

Bark 尺度 237, 238, 278, 279, 284
巴里特鲁·阿克斯 289
邦·乔维 182
包络跟随器 237, 238, 251, 284
保罗·昆斯 221, 287
保罗·兰斯基 50, 181, 246, 273, 287
贝塞尔函数 74-77, 282
《奔腾的河流》 64, 288
比利·乔 13
拨弦 162, 164-166, 168, 182, 288
波表合成 29, 30, 50, 266, 289
波形整形 51, 55, 56, 58-61, 168, 289
博德图 96, 103, 104
不变性 84-86
布莱克曼窗 208, 289

C

采样 9-16, 18, 21-24, 26, 28-34, 36-38, 40-48, 50, 53, 54, 61, 71-73, 77, 81-83, 85, 86, 88, 97-99, 103, 105, 107-109, 111, 113, 114, 133, 148, 151, 153, 157, 165, 168, 171, 173, 177, 179, 184, 185, 187-189, 191, 193, 195-197, 199-201, 203, 206, 208, 210, 215, 216, 218, 232, 242, 243, 245, 246, 252, 258, 259, 262-265, 280, 286, 289
采样点、样本 289
采样定理 12, 13, 289

采样率 9, 10, 12-16, 18, 28, 31, 40, 41, 46, 53, 68, 71, 72, 98, 136, 187, 188, 193, 199, 200, 289
采样率变换 44, 289
采样频率(f_s) 289
差分方程 62, 81, 84, 88-94, 97, 98, 103, 105, 106, 108, 113-116, 118, 120, 125-127, 131, 146, 149-153, 162, 163, 165, 167, 168, 170, 173, 174, 230, 231, 237
长除法 126, 127, 287
长时平均频谱 224, 283, 287
初始相位 251, 289
传递函数 55, 56, 59, 61, 105, 108-110, 115, 116, 118, 119, 127, 128, 146, 149-152, 175, 246, 247, 249, 289
窗 31-35, 38, 40, 47-49, 52-54, 81-83, 133, 146, 147, 187-191, 193-196, 203-210, 212, 213, 215, 224, 225, 229, 237, 240-243, 245, 248, 249, 258, 259, 261, 262, 264, 265, 281, 283, 290
窗口化 32, 290
磁带饱和度失真 58, 93, 289

D

达夫特·朋克 273
大卫·林奇 80, 287
带通滤波器 134, 135, 140, 147-149, 151-155, 181, 236-239, 250, 252-254, 256, 258, 282
带阻滤波器 113, 133-135, 141, 149
丹尼斯·枷柏 51
单位圆 1, 2, 116-118, 121, 122, 124-126, 150, 151, 153, 158, 166, 249, 251, 289
倒频 233, 288

倒谱 230, 233, 234, 284
 《盗版音乐音乐是寻找创作素材的特权》 25
 等功率平移 61, 62
 低频能量率 243, 284, 287
 低频能量率 LF 287
 低频振荡器 63, 171, 287
 低通滤波器 105, 106, 133, 134, 136, 137, 142, 149, 165, 166, 168, 181, 199, 201, 202, 224, 232, 237, 243, 248, 254, 256, 257, 282, 287
 狄利克雷内核 207, 210, 212
 《电话》 79, 288
 电台司令乐队 49, 50, 288
 《电子之诗》 63, 288
 调制模式 288
 动态扩展器 58
 动态压缩器 56, 57
 抖动 22-25, 161, 217, 218, 267, 281, 283
 短时傅立叶变换(STFT) 289
 对称性 98, 99, 102, 103, 220
 多路滤波器 170, 288

F

反 z 变换 287
 反滤波 233, 287
 反梳状滤波器 230-233, 284, 287
 《梵歌》 80, 287
 方波 21-24, 57, 60, 147, 184-190, 192, 193, 217, 218
 非因果 83, 120-126, 288
 飞利浦 13, 63, 288
 飞利浦展馆 288
 分贝(dB) 4
 分数延迟 62, 147, 168, 171-174, 176
 弗莱彻-曼森曲线 7
 幅值 5, 18, 19, 21, 23, 24, 32, 34, 36, 44, 45, 52, 55, 56, 58, 62, 68, 69, 71, 73, 75-77, 87, 88, 90-93, 96-99, 101-103, 105, 106, 108, 113, 116-119, 130, 131, 134, 136-141, 147,

148, 156, 157, 166, 171, 174-176, 178, 184, 186, 189, 190, 192, 196, 197, 200, 203, 205-207, 210-212, 217, 219, 220, 223-227, 229, 232, 234, 235, 237, 243, 249-251, 253, 256-258, 261-263, 266-268, 272, 280, 282
 《父亲》 273
 复合因果性 288
 复数 65-68, 101, 107, 116, 119, 120, 150, 192, 252, 256, 260
 傅立叶变换 13, 22, 44, 65, 73, 101, 184, 186-190, 194, 195, 197, 200, 202, 203, 213, 217-221, 224, 230, 234, 251, 256, 258, 263, 264, 283

G

《感官诱惑》 273, 289
 高通滤波器 21, 134, 138, 148, 149, 231, 232, 243, 248, 287
 格雷厄姆·贝尔 4
 光谱图 218-220, 283, 289
 归一化频率 103, 288
 滚降 203, 206-208, 268, 288
 过采样 9, 13, 44-48, 53, 54, 97, 133, 199, 201, 202, 258, 262, 281, 283
 过零率 36-38, 240-242, 281, 284, 290

H

δ 函数 87, 88, 91, 130
 汉恩窗 31-34, 47, 52, 53, 203, 207, 249, 290
 汉明窗 207, 208, 290
 合声 51, 55, 62, 63, 81, 169-174, 238, 282
 和与差 289
 亨利·科维尔 182
 哼唱检索 269, 270, 288
 后掩蔽 8
 互相关 44
 环形调制 68, 70-73, 80, 288
 黄金分割 79
 回到苏联 223

回声 62, 63, 173

混响器 63, 92, 95, 177, 180, 181, 283,
285, 288

J

《机关枪》 131, 287

《机器停止》 26, 287, 289

基频计算 36, 43, 234

激励时间 56, 57, 268-270

吉米·亨德里克斯 106, 131, 149, 181, 287

极、极点 288

加法合成 184-187, 189, 192, 217, 221, 236,
250, 251, 253, 283

减采样 13, 16, 44-48, 53, 54, 133, 199-201,
229, 234, 235, 258, 262, 265, 281, 286

减法合成 147, 148, 184, 282, 289

减 / 过采样 281

节点 93, 162, 163, 288

节拍 269, 270, 281

截止频率 133, 134, 136, 146, 148, 201, 254

矩形窗 31-34, 40, 47, 53, 188, 203-207, 210,
212, 283, 290

具像的 PH 值 63

卷积 89, 91-93, 95, 97, 179, 214, 215, 217

均方根(RMS) 28, 34, 283, 289

均匀量化 18, 288

K

卡恩海因茨·斯托克豪 289

卡普拉斯拨弦乐器模型 287

凯瑟窗 208, 209, 290

科特·利珀 95, 287

克拉拉·洛克莫尔 288

《恐怖和希望》 288

控制信号 240

快速傅立叶变换(FFT) 283

L

拉森效应 131, 287

莱德·齐柏林 79, 287

《狼人》 131, 132, 290

劳丽·安德森 273

粒子 51-55, 64, 281, 287

粒子合成 51-55, 63, 64, 287

量化 13, 17-19, 22-25, 28, 217, 218, 280,
281, 288

零点 36-39, 45, 69, 81, 114-120, 122, 127,
136, 138-144, 149-153, 158, 159, 174, 201,
202, 212, 215, 227, 246, 290

零填充 160, 195, 196, 201, 203, 204, 215,
216, 283, 290

留数 126, 129, 130, 287, 288

滤波 38, 41, 61, 63, 136, 138, 147, 149, 153,
160, 161, 167, 168, 170, 174, 177, 184, 200-
202, 229-233, 236, 237, 239, 247-250, 256,
266, 283

滤波带 237-240, 250-254, 256-258,
261, 264

滤波带组模型 253, 254, 258, 288

滤波器 8, 21, 31, 37, 38, 46, 50, 54, 63, 84,
105, 106, 109, 113, 120, 131, 133-147, 149-
153, 157, 160, 163, 165-171, 173, 174, 176,
179, 181, 186, 201, 219, 227, 229-232, 237-
239, 246-249, 251, 266, 272, 282

滤波器工具包 288

滤波器设计 136, 142, 144, 146-148, 153

滤波器设计工具 144, 148

掠夺采样 25, 288

罗伯特·阿什利 131

M

MATLAB 2, 3, 6, 12-14, 16, 21, 28, 46, 56,
92, 105, 119, 130, 131, 135, 144, 146-148,
152, 156, 160, 161, 168, 171, 180, 220, 237,
280, 283, 287

MP3 8, 269, 288

迈克尔·杰克逊 25, 287

麦当娜 181, 287

脉冲编码调制(PCM) 17, 288

脉冲响应 18, 87-93, 95, 97, 121, 125-127, 129, 131, 146, 157, 178, 179, 182, 214, 215, 217, 283, 287

《门与叹息声的变奏曲》 25, 289

米勒·帕克特 95, 288

模式(模态) 288

莫顿·费尔德曼 131

N

奈奎斯特定理 98, 99

尼克·迪克科维斯基 182

《女妖》 182, 289

O

《哦,超人》 273, 288

欧拉公式 65, 66, 68, 70, 97, 99, 100, 102, 103, 117, 150, 158, 189, 192, 197, 210, 258

P

旁频 74-78, 286, 289

披头士 223

皮埃尔·斯奇弗 25, 289

频率 1-5, 7, 8, 10, 12-17, 21, 23, 25, 28, 37, 38, 41, 43, 44, 46, 48, 52, 61, 65, 66, 68-71, 73-75, 77-79, 96-99, 101-105, 107-113, 116, 117, 133, 134, 136, 138, 140, 146, 147, 149, 151-153, 158, 160, 162, 166, 171, 174, 176, 184, 186-197, 199-205, 210-213, 215, 217-219, 221, 223, 224, 227, 230, 232-234, 236-238, 243, 245, 247-254, 256-264, 266-269, 272, 278-280, 284, 289

频率调制 66, 73, 74, 77, 78, 267, 269

频率分辨率 193-195, 218, 227, 245, 258

频率混叠 13, 16, 17, 23, 26, 46, 61, 98-101, 133, 197, 199-201, 280, 283

频率响应 89, 96-106, 114-119, 131, 137, 151, 153, 157, 158, 166-168, 174-176, 190, 197, 230, 237, 249, 282

频谱平坦量检测 244, 289

频谱通量 267, 289

频域 3, 13, 27, 34, 38, 45, 93, 101, 109, 133, 134, 136, 146, 169, 184, 186-189, 191, 193-197, 199, 201-203, 206-212, 214, 215, 217, 220, 223, 224, 226, 230, 234, 249, 254, 258, 262, 269, 270, 272, 285-287, 289, 290

平均滤波器 106, 165, 167

平均值 21, 34, 38, 81, 82, 165, 244, 288

平克·弗洛伊德乐队 223, 288

谱包络 227-229, 266, 272, 289

谱分析 184, 195, 207, 223, 224, 258, 289

谱质心 266, 267, 269, 272, 289

Q

前掩蔽 8

钱乐队,《金钱》 288

切比雪夫窗 208, 209, 290

切比雪夫多项式 59-61

曲线积分 129, 130, 287

全通滤波器 147, 171, 174-180, 282, 285, 288

群延迟 110, 111, 113, 131, 287

R

Recherche 协和声学及音乐联合研究所 (IRCAM) 287

S

声变形 54, 55, 287

声粒子 51-54, 63, 64, 289

声码器 66, 219, 223, 235, 236, 240, 246, 248-251, 253, 254, 256-259, 261, 263-265, 273, 274, 289

声强级(SIL) 4, 289

声压级(SPL) 5, 289

失真 12, 13, 16, 20-24, 26, 34, 44, 45, 49, 51, 56-58, 64, 84, 96, 98, 105, 106, 109, 111, 135, 136, 146, 168, 169, 197, 199-205, 210, 215, 218, 249, 261, 272, 281, 282

诗、诗歌 288
 《狮子的成长》 273, 287
 时长 2-4, 10, 12, 29, 30, 48, 51, 52, 54, 87, 148, 161, 169, 170, 194, 261
 时称不变性 287
 时间分辨率 38, 195
 时间离散傅立叶变换 DTFT 187, 191, 210
 时间连续 97, 107, 109, 186, 187, 201, 220
 时间延展 48, 49, 54, 258, 261-265, 289
 时间延展/压缩 287
 时域 8, 12, 13, 22, 27-29, 31, 34-36, 38, 40, 44-47, 49-51, 55, 61, 74, 87, 93, 109, 111, 126, 169, 184, 186-189, 191, 193-197, 199, 201-203, 207-209, 212, 214-217, 219, 220, 229, 230, 233, 242, 248, 249, 258, 261, 262, 264, 266, 269, 270, 272, 273, 290
 时质心 289
 史努普·多戈 273, 289
 收敛域(ROC) 288
 梳状滤波器 147, 153, 156-162, 165, 166, 168, 170, 174, 180, 181, 230-232, 282, 285, 288
 双二次曲线 149, 151, 154, 155
 双二次曲线滤波器 149
 《双峰》 80, 289
 《双曲线族沉默中的静止和运动》 289
 瞬时频率 66, 77, 250-254, 256-259, 261-264, 287
 瞬时相位 65, 66, 73, 77, 105, 250, 252, 253, 256, 258-264, 287
 《死去的机器人》 273
 《四声道》 131, 288
 索尼 13, 289

T

泰勒级数 128, 289
 泰勒级数展开 126, 128, 287
 The Who 乐队 131, 289
 汤普森兄弟 79, 289

特拉维斯·沙尔 95, 289
 听觉局限 287
 听觉掩蔽 7
 通带 133-138, 140, 142, 146, 147, 236, 288
 通道声码器 236-238, 246, 247, 250, 251
 突出峰值提取 289

W

“哇哇”颤音效果 289
 外差计算 254, 256-258, 287
 维尼·贝尔 80
 稳定性 89-91, 125, 131, 137, 150, 157, 165, 249, 287, 289
 问题与解决方法 288
 《我坐在一个房间里》 181, 287
 无限脉冲响应(IIR) 89
 无限脉冲响应 89, 146, 285, 287
 物理建模 163, 288

X

系统流程图 93, 289
 《弦乐的通道》 182, 289
 弦振动 161-163, 289
 显著特征提取 265, 266, 269, 271, 272, 289
 线性 4, 7, 28, 29, 45, 55, 56, 58, 61, 62, 66, 81, 84-87, 91, 93, 96, 103-105, 107, 109-111, 113, 124, 128, 164, 171-173, 176, 220, 225-227, 237, 246, 248, 262, 278-280, 283, 287
 线性时不变系统(LTI) 287
 线性预测编码(LPC) 284, 287, 288
 陷波滤波器 134, 153, 288
 相位 1, 2, 16, 52, 65, 66, 70, 73, 96-98, 102, 103, 105-113, 116, 119, 131, 133, 138-142, 150, 153, 156, 166, 169, 174, 176, 177, 184, 186, 187, 189, 192, 224, 249-254, 256-265, 270, 273, 282, 284, 286-288
 相位声码器 288
 相位失真 288

相位响应 288
 相位延迟 288
 相位展开 288
 相信 273
 镶边 51, 62, 63, 74, 173, 174, 213
 镶边效果器 173, 174
 歇尔 273
 谐波 22-24, 38, 49, 57, 59-61, 65, 77, 78, 162, 165, 166, 182, 184-190, 193, 205, 212, 217-219, 223-227, 230-235, 243, 247, 250, 251, 254, 266-268, 272, 278, 287
 谐波谱生成法 230, 234, 287
 谐波失真 22-25, 58, 217, 218, 221, 281, 287
 谐波失真失真 281
 信噪比 7, 19, 58, 218, 272, 288, 289
 虚数系统 287
 循环卷积 215, 216, 283

Y

压缩器 56-58, 163
 掩蔽阈限 7, 8, 287
 《夜间交通》 181, 288
 《一个人坐在咖啡厅里》 273
 伊阿尼斯·克赛纳基斯 51, 63, 290
 移动平均滤波器 43, 81, 84, 86, 106, 286, 288
 异步 53, 54, 287
 因果性 83
 音调调整 287
 音高 2-4, 25, 36-44, 46-49, 52-54, 63, 71, 74, 78, 79, 106, 157, 169, 200, 224, 227, 230-234, 238, 246, 249, 265-267, 270, 273, 278, 280, 284, 288
 音乐 3, 4, 7, 16, 19, 21, 23, 25-27, 37, 43, 46, 50, 51, 57, 58, 61-65, 68, 70, 71, 73, 78-80, 90, 91, 93, 95, 96, 106, 131-133, 142, 147, 149, 153, 160, 161, 169, 170, 181, 182, 205, 213, 218, 221-224, 230, 235, 238, 246, 249, 250, 265-267, 269-271, 273, 274, 278, 280, 285, 288

音频 3, 5, 8, 13, 18, 20, 25, 34, 46, 55, 57, 58, 61-63, 70, 71, 80, 84, 85, 90, 91, 95, 109, 133, 136, 160, 162, 214, 218, 221, 231, 240, 249, 261, 266, 269
 有限脉冲响应(FIR) 89
 《与祈祷者的生活》 182, 287
 语言声/非语言声分析 289
 语言声、语音 289
 预权重、预加权 288
 元音 219, 238, 247, 267, 268, 289
 约翰·阿普尔顿 26
 约翰·凯奇 80
 约翰·乔宁 79
 约翰·沃斯伍德 25, 288

Z

z 变换 114, 115, 290
 振荡器 2, 59-61, 63, 70-73, 77, 79, 147, 221, 288
 振幅 1-6, 11, 13, 14, 18, 19, 22, 24, 27, 30, 34, 52, 53, 56, 57, 60-62, 65, 68, 70, 71, 73, 77-79, 87, 96, 98, 105, 112, 133, 148, 162, 171, 184, 185, 187, 189, 203, 250, 266, 289
 振幅包络 14, 27-29, 31, 34, 35, 52, 53, 112, 148, 149, 161, 178, 237, 239, 266
 振幅调制 70-74, 80, 148, 219, 267, 269, 282
 振幅失真 55
 正弦 16, 20-22, 65, 68, 73, 100, 102, 108, 109, 111, 133, 158, 171, 184, 185, 189, 200, 203, 210, 244, 251, 254, 256-259, 261-264, 289
 正弦波 1, 2, 5, 6, 9-22, 26, 27, 34, 36-38, 40, 51, 52, 65, 66, 68, 73, 78, 80, 107, 184-187, 189, 196-200, 203-205, 218, 220, 221, 238, 251, 253, 254, 257, 259, 261, 262, 271, 280, 289
 正弦信号 80, 97, 186, 238, 289
 重叠 DFT 变换(ISTFT)模型 288
 重复叠加(OLA) 288

- 周期 1-3, 9-16, 25, 29, 36-38, 40, 41, 43, 48, 49, 66, 69, 97, 99, 107-109, 111, 151, 157, 162, 171, 191, 193, 197, 200, 212, 219, 240-242, 249, 252, 259, 270, 289
- 周期性 29, 40, 53, 73, 98, 102, 197, 219, 220, 238, 240, 244, 286, 288
- 《猪》 182, 288
- 属性 78, 127, 164, 219, 271
- 转换带、过渡带 289
- 自动节拍检测 270
- 自动音色识别 270
- 自相关分析 27, 40
- 阻带、止带 289
- 最小可觉差别阈限(JND) 5, 287

原著评论

朴泰弘在这本书中建立了一座连接数学声学工程世界与计算机音乐世界的坚固桥梁。无论这座桥上的旅人将会走向何方，他们都会得到清晰而富于启发性的实验。

保罗·兰斯基

普林斯顿大学“威廉·舒贝尔·科南特”音乐教授

朴泰弘博士的书以独特的跨学科视角和严谨性，介绍了音频信号处理的基本原理，带来了令人兴奋而富有深度的观点和具有创造性的应用。

艾罗罗·奥瑞芬

加利福尼亚州旧金山市音频工程师、作曲家

朴泰弘关于音乐数字信号处理的新书犹如一颗真正的宝石，每一个概念的论述都提供了许多生动的图片和容易理解的源代码。我期待着把这本书用作教材，相信学生一定会喜欢它。

格奥尔格·埃斯尔

柏林工业技术大学、德国电信实验室高级研究员



分类建议

音乐理论

ISBN 978-7-80751-808-2



定价: 48.00元